



meem by Gulf International Bank
Open Banking API Specifications

Contents

1. INTRODUCTION.....	4
2. TPP Onboarding.....	4
The Registration Process for TPP.....	4
3. Security.....	4
Mutual TLS:.....	4
JWT Signature:.....	6
4. Account Information Services (AIS) Open Banking API Specification.....	8
4.1 Accounts Consents.....	8
4.1.1 POST /account-access-consents.....	8
4.1.2 GET /account-access-consents/{ConsentId}.....	11
4.1.3 PATCH /account-access-consents/{ConsentId}.....	12
4.2 Accounts.....	13
4.2.1 GET /accounts.....	13
4.2.2 GET /accounts/{AccountId}.....	16
4.3 Balances.....	17
4.3.1 GET /accounts/{AccountId}/balances.....	17
4.4 Transactions.....	18
4.4.1. GET /accounts/{AccountId}/transactions.....	18
4.5 Beneficiaries.....	32
4.5.1 GET /accounts/{AccountId}/beneficiaries.....	32
4.6 Standing Orders.....	34
4.6.1.GET /accounts/{AccountId}/standing-orders.....	34
4.7. Sharing Product Details.....	36
4.7.1.GET /sharing-product-details.....	36
5. AIS Swagger File.....	39
6. Payment Initiation Services(PIS) Open Banking API Specification.....	40
5.1 Domestic Payments.....	41

5.1.1 Domestic Payments Consents	41
5.1.2 POST /domestic-payment-consents	41
5.1.3 GET /domestic-payment-consents/{ConsentId}.....	45
5.1.4 GET /domestic-payment-consents/{ConsentId}/funds-confirmation	46
5.2 Domestic Payments Initiation	47
5.2.1 POST /domestic-payments	47
5.2.2 GET /domestic-payments/{DomesticPaymentId}.....	49
6.3 International Payments Consents	52
5.3.1 POST /International-payment-consents	52
5.3.2 GET /international-payment-consents/{ConsentId}.....	56
5.3.3 GET /international-payment-consents/{ConsentId}/funds-confirmation	59
5.4 International Payments Initiation	59
5.4.1 POST /international-payments	59
5.4.2 GET / international-payments/{ internationalPaymentId}.....	64
7. PIS Swagger File	65

1. INTRODUCTION

GIB - Bahrain facilitates Open Banking API for the following Third-Party Providers (TPPs):

- ✦ AISP (Account Information Service Provider).
- ✦ PISP (Payment Initiation Service Provider).

2. TPP Onboarding

The Registration Process for TPP

1. Interested Account Information Service Providers & Payment Initiation Service Providers (TPPs) should contact GIB on following email (Email Address).
2. TPPs will have to furnish details such as CBB licensing certificate, contact information, Application details etc. and share with the Bank when asked.
3. TPP will have to either provide the certificates provided by authorized Qualified Trust Service Providers in the region or approach GIB with valid Certificate Signing Requests.
4. If the information shared by TPPs is verified by Bank as satisfactory, then Bank will Onboard the TPPs with Bank for Open banking.
5. Once TPPs are onboarded, the Bank will share the "client ID" Generated for TPP via email along with Instructions for Sand box testing & support contact Matrix.
6. TPP to perform sandbox testing using above document.

3. Security

TPPs Must follow the Open Banking Security standards defined by Bahrain Open Banking Framework.

<https://bahrainob.atlassian.net/wiki/spaces/BH/pages/309985587/Security+Standards+and+Guidelines>

Mutual TLS:

All requests from TPP to GIB will be using Mutual TLS mechanism to ensure transport level security. The TPP has to submit client authentication certificate signed by a trusted CA. If Qualified Trust Service Providers are not present in Bahrain then GIB CA signed certificates need to be produced.

Below are the steps for the TPP to generate Private Key and Certificate Signing Request to be used for Mutual TLS.

Sample Command:

```
openssl req -new -config qwac_config.cnf -out qwac_cert.csr -keyout qwac_private.key
```

Sample Configuration File:

(qwac_config.cnf) for Mutual TLS Client Authentication Certificate:

```
#  
# CSR REQUEST CONFIGURATION FILE  
# Mutual TLS Client Authentication Certificate Request
```

```
# -----  
# Add new_oids OID  
#  
oid_section = new_oids  
[ new_oids ]  
organizationIdentifier = "2.5.4.97" # This is recommended  
[ req ]  
default_bits = 2048 # RSA key size  
encrypt_key = yes # Protect private key: yes or no. yes recommended  
default_md = sha256 # MD to use. sha256 recommended utf8 = yes # Input is UTF-8.  
string_mask = utf8only # Emit UTF-8 strings prompt = no # Prompt for DN. yes or no.  
distinguished_name = client_dn # DN template. Mandatory to include organizationIdentifier  
req_extensions = client_reqext # Desired extensions.  
#  
# Subject Distinguished Name format in certificate  
# -----  
# EG: CN = 12345678910, 2.5.4.97 = PSDIN-OB-12345678910, O = Open Banking Limited (D), C = GB  
#  
[ client_dn ]  
countryName = "BH" # Country code - see doc above  
organizationName = "TPP (BAHRAIN) LIMITED" # Organizational name  
#  
# The organizationIdentifier shall be present in the Subject's Distinguished Name  
#  
organizationIdentifier = "PSDIN-OB-12345678910" commonName  
= "12345678910" # Subject common name  
# emailAddress = "yourname@yourorganisation.com" # Organisation email contact address  
# Required specific extensions in certificate #  
[ client_reqext ]  
keyUsage = critical,digitalSignature # Must be critical  
extendedKeyUsage = clientAuth, serverAuth # Must be defined as shown above  
subjectKeyIdentifier = hash # Hash value to calculate SKI  
#  
# qcStatements extension will specify the role of the TPP – AISP, PISP etc.  
# The given qcStatements below is for the TPP with both AISP and PISP roles  
#  
qcStatements=DER:30818e3013060604008e4601063009060704008e4601060330770606040081  
982702306d304630440607040081982701010c065053505f41530607040081982701020c0650535  
05f50490607040081982701030c065053505f41490607040081982701040c065053505f49430c1b4  
6696e616e6369616c20436f6e6475637420417574686f726974790c0647422d464341
```

JWT Signature:

The TPP has to sign the JWT assertions (client credential – client assertions) using a valid Electronic Seal certificate signed by a trusted CA. If Qualified Trust Service Providers are not present in Bahrain then then GIB CA signed certificates need to be produced.

Below are the steps for the TPP to generate Private Key and Certificate Signing Request to be used for Electronic Seal (JWT Signature).

Sample Command:

```
openssl req -new -config qseal_config.cnf -out qseal_cert.csr -keyout qseal_private.key
```

Sample Configuration File:

(qseal_config.cnf) for Electronic Seal (JWT Signature) Certificate:

```
#
# CSR REQUEST CONFIGURATION FILE
# Electronic Seal Client Certificate Request
# -----
# Add new_oids OID
#
oid_section = new_oids
[ new_oids ]
organizationIdentifier = 2.5.4.97
[ req ]
default_bits = 2048 # RSA key size
encrypt_key = yes # Protect private key: yes or no. yes recommended
default_md = sha256 # MD to use. sha256
recommended_utf8 = yes # Input is UTF-8. string_mask =
utf8only # Emit UTF-8 strings prompt = no # Prompt for
DN. yes or no.
distinguished_name = client_dn # DN template. Mandatory to include organizationIdentifier
req_extensions = client_reqext # Desired extensions.
#
# Subject Distinguished Name format in certificate
# -----
# EG: CN = 12345678910, 2.5.4.97 = PSDIN-OB-12345678910, O = Open Banking Limited (D), C =
GB
#
#
[ client_dn ]
countryName = "BH" # Country code - see doc above
organizationName = "TPP (BAHRAIN) LIMITED" # Organizational name
#
# The organizationIdentifier shall be present in the Subject's Distinguished Name
```

```
#
organizationIdentifier = "PSDIN-OB-12345678910" commonName
= "12345678910" # Subject common name
```

```
# Required specific extensions in certificate #
[ client_reqext ]
keyUsage = critical,digitalSignature,nonRepudiation # Must be critical subjectKeyIdentifier
= hash
#
# qcStatements extension will specify the role of the TPP – AISP, PISP etc.
# The given qcStatements below is for the TPP with both AISP and PISP roles
#
qcStatements=DER:30818e3013060604008e4601063009060704008e4601060330770606040081
982702306d304630440607040081982701010c065053505f41530607040081982701020c0650535
05f50490607040081982701030c065053505f41490607040081982701040c065053505f49430c1b4
6696e616e6369616c20436f6e6475637420417574686f726974790c0647422d464341
```

4. Account Information Services (AIS) Open Banking API Specification

Account Information Open Banking API provides information of all of the accounts in active status that a customer has with bank.

Following are the AIS APIs supported by GIB – Bahrain for open Banking

4.1 Accounts Consents

The Account Access Consents API is used by an AISP to request an ASPSP to create a new accountaccess-consents resource, retrieve the status of account-access-consents resource and patch the account-access-consents resource.

4.1.1 POST /account-access-consents

1. User should give consent to an AISP accessing their account information and AISP should connect to Bank's API to create an account-access-consent resource using POST /account-access-consents.
2. The account-access-consent resource includes the fields below which describe data the PSU has consented with the AISP:
 - ✦ **Permissions** - List of data clusters that have been consented by user for data access
 - ✦ Only following permissions are supported by GIB – Bahrain, and AISPs are required to send requests with only the supported permissions.
 - ✦ ReadAccountsBasic
 - ✦ ReadAccountsDetail
 - ✦ ReadBalances
 - ✦ ReadBeneficiariesBasic
 - ✦ ReadBeneficiariesDetail
 - ✦ ReadStandingOrdersBasic
 - ✦ ReadStandingOrdersDetail
 - ✦ ReadTransactionsBasic
 - ✦ ReadTransactionsCredits
 - ✦ ReadTransactionsDebits
 - ✦ ReadTransactionsDetail
 - ✦ **Expiration Date** – Date after which the AISP will no longer have access to the User's data.
 - ✦ **Transaction Validity Period** - The To / From date range specifying historical period for transactions which can be accessed by the AISP.
2. This informs Bank that their customers is granting access to account information to an AISP and Bank responds with an identifier for the resource (ConsentId) and redirection URL.
3. The AISP should redirect the customer to GIB- Bahrain's Login page and bank will use two factor authentications to authenticate the customer. During authorization, the customer must select accounts which customer wishes to grant access to AISP.
4. Once the consent has been authorized/declined, the customer will be redirected back to the AISP's page with an auth code.
5. TPP has to retrieve user access token using the auth code

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

POST <https://apis.meem.bh/token>

Mandatory field(s)

Authorization =Application_Access_Token

Body =

```
{
  "Data":{
    "Permissions": [
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadBeneficiariesDetail",
      "ReadStandingOrdersDetail",
      "ReadTransactionsCredits",
      "ReadTransactionsDebits",
      "ReadTransactionsDetail",
    ],
    "ExpirationDateTime":"2022-10-11T10:28:49.599+05:30",
    "TransactionFromDateTime":"2020-10-06T10:28:49.600+05:30",
    "TransactionToDateTime":"2020-10-09T10:28:49.600+05:30"
  },
  "Risk":{
  }
}
```

Sample Response

```
{
  "Meta": {
    "TotalPages": 1
  },
  "Links": {
    "Self": "https://10.8.9.131:8243/open-banking/v3.1/aisp/account-access-consents/0396ac51-8743-43c4-9ec6b4cbba463e2a"
  },
  "Data": {
    "Status": "Authorised",
    "StatusUpdateDateTime": "2021-04-23T14:23Z",
    "CreationDateTime": "2021-04-23T14:18Z",
    "TransactionToDateTime": "2021-06-06T10:28:49+05:30",
    "Permissions": [
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadBeneficiariesDetail",
      "ReadStandingOrdersDetail",
      "ReadTransactionsCredits",
      "ReadTransactionsDebits",
      "ReadTransactionsDetail"
    ],
    "ConsentId": "0396ac51-8743-43c4-9ec6-b4cbba463e2a",
    "TransactionFromDateTime": "2015-01-04T10:28:49+05:30"
  }
}
```

4.1.2 GET /account-access-consents/{ConsentId}

This API may be used by AISP to get details and status of any specific ConsentId provided by Bank.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/account-access-consents>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```
{
  "Meta": {
    "TotalPages": 1
  },
  "Links": {
    "Self": "https://10.8.9.131:8243/open-banking/v3.1/aisp/account-access-consents/0396ac51-8743-43c4-9ec6b4cbb463e2a"
  },
  "Data": {
    "Status": "Authorised",
    "StatusUpdateDateTime": "2021-04-23T14:23Z",
    "CreationDateTime": "2021-04-23T14:18Z",
    "TransactionToDateTime": "2021-06-06T10:28:49+05:30",
    "Permissions": [
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadBeneficiariesDetail",
      "ReadStandingOrdersDetail",
    ]
  }
}
```

```

    "ReadTransactionsCredits",
    "ReadTransactionsDebits",
    "ReadTransactionsDetail",
  ],
  "ConsentId": "0396ac51-8743-43c4-9ec6-b4cbba463e2a",
  "TransactionFromDate": "2015-01-04T10:28:49+05:30"
}
}

```

4.1.3 PATCH /account-access-consents/{ConsentId}

This API may be used by AISP to update the status of any specific ConsentId to revoked. Once Any consent status is changed to Revoked, any subsequent request related to that consent ID will return unauthorized response.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

PATCH <https://apis.meem.bh/open-banking/v3.1/aisp/account-access-consents/{ConsentId}>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```

{
  "Meta": {
    "TotalPages": 1
  },
  "Links": {
    "Self": "https://10.8.9.131:8243/open-banking/v3.1/aisp/account-access-consents/0396ac51-8743-43c4-9ec6b4cbba463e2a"
  },
  "Data": {

```

```

    "Status": "Revoked",
    "StatusUpdateDateTime": "2021-04-23T14:30Z",
    "CreationDateTime": "2021-04-23T14:18Z",
    "TransactionToDateTime": "2021-06-06T10:28:49+05:30",
    "Permissions": [
        "ReadAccountsDetail",
        "ReadBalances",
        "ReadBeneficiariesDetail",
        "ReadStandingOrdersDetail",
        "ReadTransactionsCredits",
        "ReadTransactionsDebits",
        "ReadTransactionsDetail",
    ],
    "ConsentId": "0396ac51-8743-43c4-9ec6-b4cbba463e2a",
    "TransactionFromDateTime": "2015-01-04T10:28:49+05:30"
}
}

```

4.2 Accounts

After successful authorization of account access consent, an AISP may retrieve account information using the following API End points

4.2.1 GET /accounts

First step after successful authorization of account access consent by AISP is to call this GET/accounts API to retrieve details for all authorized accounts (AccountID) linked to the account access consent. These Account IDs must be used for other AIS APIs.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts>

Mandatory field(s)

Authorization =User_Access_Token

Sample Response

```
{
  "Data": {
    "Account": [
      {
        "AccountId": "CA3XXXXXXXX48",
        "Status": "Enabled",
        "Currency": "BHD",
        "AccountType": "Personal",
        "AccountSubType": "CurrentAccount",
        "Description": "Current Account",
        "OpeningDate": "2020-08-09T00:00:00",
        "Account": [
          {
            "SchemeName": "BH.OBF.IBAN",
            "Identification": "BH54GULF0003XXXXXXXX48",
            "Name": "SXXXXXXXX X"
          }
        ],
      },
      {
        "SchemeName": "BH.OBF.BICFI",
        "Identification": "GULFBHBM399"
      }
    ],
    "Servicer": {
      "SchemeName": "BH.OBF.BICFI",
      "Identification": "GULFBHBM399"
    }
  },
  {
    "AccountId": "FD3XXXXXXXX9",
    "Status": "Enabled",
    "Currency": "BHD",
    "AccountType": "Personal",
```

```
"AccountSubType": "Deposit",
>Description": "Deposit",
>MaturityDate": "2021-05-11T00:00:00",
>Account": [
>  {
>    "SchemeName": "BH.OBF.IBAN",
>    "Identification": "3XXXXXXXX9",
>    "Name": null
>  }
>]
},
{
>AccountId": "CC3b056c44-6316-4cbd-b954-f1c32625a492",
>Status": "Enabled",
>Currency": "BHD",
>AccountType": "Personal",
>AccountSubType": "CreditCard",
>Description": "Credit Card",
>OpeningDate": "2020-05-11T00:00:00",
>Account": [
>  {
>    "SchemeName": "BH.OBF.PAN",
>    "Identification": "46XXXXXXXX2",
>    "Name": "SXXXXXXXX X"
>  }
>]
}
],
>Links": {
>Self": "http://www.gib.com/accounts"
}
}
}
```

4.2.2 GET /accounts/{AccountId}

This API may be called by AISPs to retrieve details for any specific AccountID linked to the account access consent

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountId}>

Mandatory field(s)

Authorization =User_Access_Token

AccountID = Unique identification as assigned by the Bank to uniquely identify the account access consent resource.

Sample Response

```
{
  "Data": {
    "Account": [
      {
        "AccountId": "CXXXXXXXX8",
        "Status": "Enabled",
        "Currency": "BHD",
        "AccountType": "Personal",
        "AccountSubType": "CurrentAccount",
        "Description": "Current Account",
        "OpeningDate": "2020-08-09T00:00:00",
        "Account": [
          {
            "SchemeName": "BH.OBF.IBAN",
            "Identification": "BHXXXXXXXXXXXXXXXX48",
```



```

        "Name": "Selvanathan K"
    }
],
"Servicer": {
    "SchemeName": "BH.OBF.BICFI",
    "Identification": "GUXXXXXX9"
}
}
],
"Links": {
    "Self": "http://www.gib.com"
}
}

```

4.3 Balances

After successful authorization of account access consent, an AISP may retrieve Balance information for a specific AccountID using the following API End point.

4.3.1 GET /accounts/{AccountID}/balances

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountID}/balances>

Mandatory field(s)

Authorization =User_Access_Token

AccountID = Unique identification as assigned by the Bank to uniquely identify the account access consent resource.

Sample Response

```

{
  "Data": {
    "Balance": [
      {
        "AccountID": "CXXXXXXXX8",

```

```

    "CreditDebitIndicator": "Credit",
    "Type": "ClosingAvailable",
    "DateTime": "2021-04-23T12:35:42.311+03:00",
    "Amount": {
      "Amount": "200047.942",
      "Currency": "BHD"
    }
  }
],
"Links": {
  "Self": "http://www.gib.com"
}
}}

```

4.4 Transactions

After successful authorization of account access consent, an AISP may retrieve list of transactions for a specific AccountID using the following API End point. Transaction Period may be included in the request.

Pagination will be used for this API if number of records to be returned is more.

4.4.1. GET /accounts/{AccountID}/transactions

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountID}/transactions>

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountID}/transactions?pg=2>

Mandatory field(s)

Authorization =User_Access_Token

AccountID = Unique identification as assigned by the Bank to uniquely identify the account access consent resource.

Sample Response

```
{
```

```
"Data": {
  "Transaction": [
    {
      "AccountId": "CXXXXXXXXX8",
      "CreditDebitIndicator": "Credit",
      "BookingDateTime": "2021-04-22 T 21:31:18",
      "ValueDateTime": "2021-04-22 T 21:31:18",
      "Status": "Booked",
      "Amount": {
        "Amount": "6.210",
        "Currency": "BHD"
      },
      "TransactionInformation": "Standing Instruction Credit\nReference : 399XXXXXXXXX01\nRemitter Number: SXXXXXXXXX\nRemitter Name: PO BOX XXXXXXXX RD XXXXXXXX \nAmount : BHD 6.210\nExchange Rate : 1\nCharge Amount : 0.00\nBranch 399",
      "Balance": {
        "CreditDebitIndicator": "Credit",
        "Amount": {
          "Amount": "996.895",
          "Currency": "BHD"
        }
      }
    },
    {
      "AccountId": "CXXXXXXXXX8",
      "CreditDebitIndicator": "Debit",
      "BookingDateTime": "2021-04-22 T 07:30:54",
      "ValueDateTime": "2021-04-22 T 07:30:54",
      "Status": "Booked",
      "Amount": {
        "Amount": "0",
        "Currency": "BHD"
      }
    }
  ]
}
```

"TransactionInformation": "Fawri Payment\nReference :399XXXXXXXXXXXXXXXX0002\nBeneficiary Number:
/Bh86aliXXXXXXXX1234\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal Ops
Code 5XXXXXXXX01\nFresco-XXXXXXXX1",

"Balance": {
 "CreditDebitIndicator": "Debit",
 "Amount": {
 "Amount": "990.685",
 "Currency": "BHD"
 }
}

},

{

 "AccountId": "CXXXXXXXXX8",
 "CreditDebitIndicator": "Debit",
 "BookingDateTime": "2021-04-22 T 02:57:56",
 "ValueDateTime": "2021-04-22 T 02:57:56",
 "Status": "Booked",
 "Amount": {
 "Amount": "0",
 "Currency": "BHD"
 }
},

"TransactionInformation": "Fawri Payment\nReference :399XXXXXXXXX01\nBeneficiary Number:
/Bh86aXXXXXXXX34\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal Ops Code
5XXXXXXXX1\nFresco-1XXXXXXXX1",

"Balance": {
 "CreditDebitIndicator": "Debit",
 "Amount": {
 "Amount": "993.790",
 "Currency": "BHD"
 }
}

},

{

 "AccountId": "CXXXXXXXXX8",

```
"CreditDebitIndicator": "Debit",
"BookingDateTime": "2021-04-23 T 12:14:34",
"ValueDateTime": "2021-04-23 T 12:14:34",
>Status": "Booked",
Amount": {
  Amount": "0",
  Currency": "BHD"
},
TransactionInformation": "Fawri Payment\nReference :39XXXXXXXXX01\nBeneficiary Number:
/Bh54guXXXXXXXXX948\nBeneficiary Name: SXXXXXXXXXN K\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal Ops
Code 5XXXXXXXXX1\nFresco-1XXXXXXXXX1",
Balance": {
  CreditDebitIndicator": "Debit",
  Amount": {
    Amount": "996.895",
    Currency": "BHD"
  }
},
},
{
  AccountId": "CXXXXXXXXXX8",
  CreditDebitIndicator": "Credit",
  BookingDateTime": "2021-04-21 T 18:21:16",
  ValueDateTime": "2021-04-21 T 18:21:16",
  Status": "Booked",
  Amount": {
    Amount": "9.105",
    Currency": "BHD"
  }
},
TransactionInformation": "Standing Instruction Credit\nReference : 399XXXXXXXXX01\nRemitter Number:
SXXXXXXXXXN\nRemitter Name: PO BOX XXXXXXXX RD XXXXXXXX 4\nAmount : BHD 9.105\nExchange Rate : 1\nCharge Amount :
0.00\nBranch 399",
Balance": {
  CreditDebitIndicator": "Credit",
```

```
"Amount": {
  "Amount": "1000.000",
  "Currency": "BHD"
}
},
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-21 T 16:05:42",
  "ValueDateTime": "2021-04-21 T 16:05:42",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  },
  "TransactionInformation": "Fawri Payment\nReference :399XXXXXXXXX001\nBeneficiary Number:
/Bh86XXXXXXXX1234\nBeneficiary Name: SXXXXXXXXN K\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal Ops
Code 512XXXXXXXXX01\nFresco-1XXXXXXXXX1",
  "Balance": {
    "CreditDebitIndicator": "Debit",
    "Amount": {
      "Amount": "990.895",
      "Currency": "BHD"
    }
  }
},
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-21 T 20:16:04",
  "ValueDateTime": "2021-04-21 T 20:16:04",
  "Status": "Booked",
```

```
"Amount": {
  "Amount": "0",
  "Currency": "BHD"
},
  "TransactionInformation": "Internal Transfer (Account to Account)\nReference : 39XXXXXXXX03\nBeneficiary Number: /Bh86guXXXXXXXX4\nBeneficiary Name: XXXXXXXX\nGTB Channel Transaction",
  "Balance": {
    "CreditDebitIndicator": "Debit",
    "Amount": {
      "Amount": "994.000",
      "Currency": "BHD"
    }
  }
},
{
  "AccountId": "XXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-21 T 20:10:36",
  "ValueDateTime": "2021-04-21 T 20:10:36",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  }
},
  "TransactionInformation": "Internal Transfer (Account to Account)\nReference : 39XXXXXXXX002\nBeneficiary Number: /Bh86gulfXXXXXXXXXXXXXXXX54\nBeneficiary Name: XXXXXXXX\nGTB Channel Transaction",
  "Balance": {
    "CreditDebitIndicator": "Debit",
    "Amount": {
      "Amount": "997.000",
      "Currency": "BHD"
    }
  }
}
```

},

{

"AccountId": "CXXXXXXXXXX8",
"CreditDebitIndicator": "Credit",
"BookingDateTime": "2021-04-20 T 18:22:23",
"ValueDateTime": "2021-04-20 T 18:22:23",
"Status": "Booked",
"Amount": {
 "Amount": "6.105",
 "Currency": "BHD"
},

"TransactionInformation": "Standing Instruction Credit\nReference : 399SIOPXXXXXXXXX001\nRemitter Number: SEXXXXXXXX N\nRemitter Name: PXXXXXXXX XXXXXXXX23 XXXXXXXX4\nAmount : BHD 6.105\nExchange Rate : 1\nCharge Amount : 0.00\nBranch 399",

"Balance": {
 "CreditDebitIndicator": "Credit",
 "Amount": {
 "Amount": "1000.000",
 "Currency": "BHD"
 }
}

}

},

{

"AccountId": "CXXXXXXXXXX8",
"CreditDebitIndicator": "Debit",
"BookingDateTime": "2021-04-20 T 16:18:41",
"ValueDateTime": "2021-04-20 T 16:18:41",
"Status": "Booked",
"Amount": {
 "Amount": "0",
 "Currency": "BHD"
},

"TransactionInformation": "Fawri Plus Payment\nReference :399FXXXXXXXXX001\nBeneficiary Number:

/Bh86alir012XXXXXXXXX34\nBeneficiary Name: XXXXXXXX Inc\nPurpose of Payment : Internal Ops Code
512XXXXXXXXX01\nFresco-10XXXXXXXXX1\n",

```
"Balance": {  
  "CreditDebitIndicator": "Debit",  
  "Amount": {  
    "Amount": "993.895",  
    "Currency": "BHD"  
  }  
}  
},  
{  
  "AccountId": "CXXXXXXXXXX8",  
  "CreditDebitIndicator": "Debit",  
  "BookingDateTime": "2021-04-20 T 14:52:57",  
  "ValueDateTime": "2021-04-20 T 14:52:57",  
  "Status": "Booked",  
  "Amount": {  
    "Amount": "0",  
    "Currency": "BHD"  
  },
```

"TransactionInformation": "Fawri Payment\nReference :399FWXXXXXXXXX006\nBeneficiary Number:
/Bh86alXXXXXXXXX7891234\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal
Ops Code 512XXXXXXXXX1\nFresco-101XXXXXXXXX1",

```
"Balance": {  
  "CreditDebitIndicator": "Debit",  
  "Amount": {  
    "Amount": "996.895",  
    "Currency": "BHD"  
  }  
}  
},  
{  
  "AccountId": "CXXXXXXXXXX8",  
  "CreditDebitIndicator": "Credit",
```

"BookingDateTime": "2021-04-19 T 18:21:27",

"ValueDateTime": "2021-04-19 T 18:21:27",

"Status": "Booked",

"Amount": {

"Amount": "17.223",

"Currency": "BHD"

},

"TransactionInformation": "Standing Instruction Credit\nReference : 399SIOP202220001\nRemitter Number: SXXXXXXXXN\nRemitter Name: PO BOX XXXXXXXX 23 RDXXXXXXXXXLK 4\nAmount : BHD 17.223\nExchange Rate : 1\nCharge Amount : 0.00\nBranch 399",

"Balance": {

"CreditDebitIndicator": "Credit",

"Amount": {

"Amount": "1000.000",

"Currency": "BHD"

}

}

},

{

"AccountId": "CXXXXXXXXX8",

"CreditDebitIndicator": "Debit",

"BookingDateTime": "2021-04-19 T 13:29:57",

"ValueDateTime": "2021-04-19 T 13:29:57",

"Status": "Booked",

"Amount": {

"Amount": "0",

"Currency": "BHD"

},

"TransactionInformation": "Outward International Transfer\nReference : 399XXXXXXXX90002\nBeneficiary Number: 08080021325698/\nBeneficiary Name: XXXXXXXXInc\nBeneficiary Bank : XXXXXXXXBank Of XXXXXXXXS.A.K.P.\nAmount : KWD 7.000\nExchange Rate : 1.266845457593\nCharge Amount : BHD 5.000\nVAT Amount : BHD 0.250\nGTB Channel Transaction",

"Balance": {

"CreditDebitIndicator": "Debit",

"Amount": {

```
"Amount": "982.777",
  "Currency": "BHD"
}
},
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-19 T 12:43:26",
  "ValueDateTime": "2021-04-19 T 12:43:26",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  },
  "TransactionInformation": "Fawri Payment\nReference :399XXXXXXXXX001\nBeneficiary Number:
/Bh86aIXXXXXXXXX91234\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\nPurpose of Payment : Internal Ops
Code 5XXXXXXXXX101\nFresco-\nFresXXXXXXXX",
  "Balance": {
    "CreditDebitIndicator": "Debit",
    "Amount": {
      "Amount": "996.895",
      "Currency": "BHD"
    }
  }
},
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Credit",
  "BookingDateTime": "2021-04-16 T 18:21:18",
  "ValueDateTime": "2021-04-16 T 18:21:18",
  "Status": "Booked",
  "Amount": {
```

```
"Amount": "106.315",
"Currency": "BHD"
},
"TransactionInformation": "Standing Instruction Credit\nReference : 399XXXXXXXX001\nRemitter Number:
XXXXXXXXXAN\nRemitter Name: XXXXXXXXFLAXXXXXXXXX 2XXXXXXXXBLK 4\nAmount : BHD 106.315\nExchange Rate :
1\nCharge Amount : 0.00\nBranch 399",
"Balance": {
  "CreditDebitIndicator": "Credit",
  "Amount": {
    "Amount": "1000.000",
    "Currency": "BHD"
  }
}
},
{
  "AccountId": "CXXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-16 T 13:58:23",
  "ValueDateTime": "2021-04-16 T 13:58:23",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  }
},
"TransactionInformation": "Fawri Payment\nReference :39XXXXXXXX60511\nBeneficiary Number:
/Bh8XXXXXXXX1234\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\n",
"Balance": {
  "CreditDebitIndicator": "Debit",
  "Amount": {
    "Amount": "893.685",
    "Currency": "BHD"
  }
}
},
}
```

```
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-16 T 10:18:11",
  "ValueDateTime": "2021-04-16 T 10:18:11",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  },
  "TransactionInformation": "Fawri Payment\nReference :399XXXXXXXXX02\nBeneficiary Number:
/Bh86XXXXXXXXX234\nBeneficiary Name: XXXXXXXX nc\nCharge Amount : BHD 0.100\n",
  "Balance": {
    "CreditDebitIndicator": "Debit",
    "Amount": {
      "Amount": "923.790",
      "Currency": "BHD"
    }
  }
},
{
  "AccountId": "CXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-16 T 07:52:48",
  "ValueDateTime": "2021-04-16 T 07:52:48",
  "Status": "Booked",
  "Amount": {
    "Amount": "0",
    "Currency": "BHD"
  },
  "TransactionInformation": "Fawri Payment\nReference :39XXXXXXXXX6XXXXXXXXX1\nBeneficiary Number:
/Bh86aXXXXXXXXXrXXXXXXXXX91234\nBeneficiary Name: XXXXXXXX Inc\nCharge Amount : BHD 0.100\n",
  "Balance": {
```

```
"CreditDebitIndicator": "Debit",

"Amount": {
  "Amount": "961.895",
  "Currency": "BHD"
}
},
{
  "AccountId": "CXXXXXXXXXX8",
  "CreditDebitIndicator": "Credit",
  "BookingDateTime": "2021-04-15 T 18:23:27",
  "ValueDateTime": "2021-04-15 T 18:23:27",
  "Status": "Booked",
  "Amount": {
    "Amount": "39.105",
    "Currency": "BHD"
  },
  "TransactionInformation": "Standing Instruction Credit\nReference : 3r012XXXXXXXXX101\nRemitter Number:
SXXXXXXXXN\nRemitter Name: PO XXXXXXXX XXXXXXXX3 RD 3453 BLK 4\nAmount : BHD 39.105\nExchange Rate : 1\nCharge
Amount : 0.00\nBranch 399",
  "Balance": {
    "CreditDebitIndicator": "Credit",
    "Amount": {
      "Amount": "1000.000",
      "Currency": "BHD"
    }
  }
},
{
  "AccountId": "CXXXXXXXXXX8",
  "CreditDebitIndicator": "Debit",
  "BookingDateTime": "2021-04-15 T 17:32:05",
  "ValueDateTime": "2021-04-15 T 17:32:05",
```

```
"Status": "Booked",

"Amount": {
  "Amount": "0",
  "Currency": "BHD"
},

"TransactionInformation": "Fawri Payment\nReference :39XXXXXXXXXX04\nBeneficiary Number:
/Bh86alir0XXXXXXXX1234\nBeneficiary Name: ACME Inc\nCharge Amount : BHD 0.100\n",

"Balance": {
  "CreditDebitIndicator": "Debit",
  "Amount": {
    "Amount": "960.895",
    "Currency": "BHD"
  }
}
},

"Links": {
  "Self": "http://www.gib.com/accounts/CXXXXXXXXX8/transactions",
  "First": "http://www.gib.com/accounts/CXXXXXXXXX8/transactions?pg=1",
  "Last": "http://www.gib.com/accounts/CXXXXXXXXX8/transactions?pg=4"
},

"Meta": {
  "TotalPages": "4"
}
}
```

4.5 Beneficiaries

After successful authorization of account access consent, an AISP may retrieve list of beneficiaries for a specific AccountID using the following API End point.

4.5.1 GET /accounts/{AccountID}/beneficiaries

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountID}/beneficiaries>

Mandatory field(s)

Authorization =User_Access_Token

AccountID = Unique identification as assigned by the Bank to uniquely identify the account access consent resource.

Sample Response

```
{
  "Data": {
    "Beneficiary": [
      {
        "AccountId": "CXXXXXXXXX8",
        "BeneficiaryId": "1XXXXXXXXX",
        "BeneficiaryType": "Ordinary",
        "Reference": "ITU",
        "CreditorAgent": {
          "Name": null,
          "PostalAddress": {
            "AddressLine": null,
            "TownName": null,
            "Country": null
          }
        },
        "CreditorAccount": {
          "SchemeName": "BH.OBF.IBAN",

```



```
    "Identification": "4XXXXXXXXX4",
    "Name": "XXXXXXXXX Test User"
  }
},
{
  "AccountId": "CA39900020948",
  "BeneficiaryId": "12517",
  "BeneficiaryType": "Ordinary",
  "Reference": null,
  "CreditorAgent": {
    "Name": null,
    "PostalAddress": {
      "AddressLine": null,
      "TownName": null,
      "Country": null
    }
  },
  "CreditorAccount": {
    "SchemeName": "BH.OBF.IBAN",
    "Identification": "BHXXXXXXXXXXXX4",
    "Name": "XXXXXXXXXX XXXXXXXXXX"
  }
},
{
  "Links": {
    "Self": "http://www.gib.com"
  }
}
}
```

4.6 Standing Orders

After successful authorization of account access consent, an AISP may retrieve list of standing orders for a specific AccountID using the following API End point.

4.6.1.GET /accounts/{AccountID}/standing-orders

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/accounts/{AccountID}/standing-orders>

Mandatory field(s)

Authorization =User_Access_Token

AccountID = Unique identification as assigned by the Bank to uniquely identify the account access consent resource.

Sample Response

```
{
  "Data": {
    "StandingOrder": [
      {
        "AccountID": "CXXXXXXXXX8",
        "Frequency": "IntrvlMnthDay:01:01",
        "Reference": "399XXXXXXXXX2",
        "NextPaymentDateTime": "0001-01-01T00:00:00",
        "NumberOfPayments": 2,
        "NextPaymentAmount": {
          "Amount": 0,
          "Currency": "BHD"
        }
      }
    ],
    {
      "AccountID": "CXXXXXXXXX8",
```

```
"Frequency": "IntrvlMnthDay:01:01",
"Reference": "399FXXXXXXXXXX015",
"NextPaymentDateTime": "0001-01-01T00:00:00",
"NumberOfPayments": 2,
"NextPaymentAmount": {
  "Amount": 0,
  "Currency": "BHD"
}
}
},
"Links": {
  "Self": "http://www.gib.com"
}
}
}
```

4.7. Sharing Product Details

This API allows AISPs to request for GIB-Bahrain's products details. If Requested Bank will return details about products and services offered by Bank.

4.7.1.GET /sharing-product-details

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/aisp/sharing-product-details>

Mandatory field(s)

Authorization =Application_Access_Token

Sample Response

```
{
  "Data": {
    "CreditCards": [
      {
        "ProductDetails": {
          "CategoryName": "meem VISA platinum Credit Card",
          "CategoryCode": "CC",
          "ProductDescription": "meem VISA platinum Credit Card",
          "ProductCode": 1001,
          "SubProductCode": 1001,
          "ProductURL": "https://bh.meem.com/credit-card/",
          "Segment": "Retail"
        }
      }
    ],
    "CASA": [
      {
        "ProductDetails": {
```

```
"CategoryName": "CurrentAccount SavingsAccount",
"CategoryCode": "CASA",
"ProductDescription": "combines the Current Account and Savings",
"ProductCode": 1002,
"SubProductCode": 1002,
"ProductURL": "https://bh.meem.com/credit-card/",
"Segment": "Retail"
}
}
],
"Investments": [
{
"ProductDetails": {
"CategoryName": "Murabaha Deposit",
"CategoryCode": "IA",
"ProductDescription": "A low risk investment with available in 8 different currencies",
"ProductCode": 1003,
"SubProductCode": 1003,
"ProductURL": "https://bh.meem.com/Murabaha-Deposit/",
"Segment": "Retail"
}
}
],
"FundTransfer": [
{
"ProductDetails": {
"CategoryName": "Fund transfer",
"CategoryCode": "FT",
"ProductDescription": "Transfer funds between onepack account and currencies accounts",
"ProductCode": 1004,
"SubProductCode": 1004,
"ProductURL": "https://bh.meem.com/Murabaha-Deposit/",
"Segment": "Retail"
}
}
]
```

```
    },
    "Features": {
      "Currency": "BHD",
      "MinFundTransfer": 10,
      "HandlingFeesSupplementary": 50,
      "TypesOfFundTransfer": "1.Fawri 2.Fawri+ 3.Fawteer"
    }
  }
],
"Loans": [
  {
    "ProductDetails": {
      "CategoryName": "Loans",
      "CategoryCode": "LN",
      "ProductDescription": "Home Loan and Personal Loan",
      "ProductCode": 1005,
      "SubProductCode": 1005,
      "ProductURL": "https://bh.meem.com/loans/",
      "Segment": "Retail"
    },
    "Features": {
      "Currency": "BHD",
      "InitialDeposit": 100,
      "HandlingFeesSupplementary": 50,
      "LoanTenure": 48,
      "InterestRates": "8% inclusive of taxes"
    }
  }
]
}
```

5. AIS Swagger File

Please Find the swagger files for PIS



account-info-swagger-3.1.5.yaml

6. Payment Initiation Services(PIS) Open Banking API Specification

GIB -Bahrain supports following payment types via Open banking. Customers will maintain full control for such payments initiated from TPPs portal by authenticating each payment on Bank's Portal.

Domestic Payments

Fawri

Fawri+

International Payments

SWIFT

5.1 Domestic Payments

5.1.1 Domestic Payments Consents

Every domestic Payment request must be preceded with Payment consents, and this Payment consent must be authorized by the customer at GIB Bahrain page.

5.1.2 POST /domestic-payment-consents

1. User should give consent to a PISP to make a payment on behalf a customer, after this the PISP uses this API to create a payment-consent resource
2. The domestic payment consent resource includes all the required payment related fields which are consented by the user with PISP. Below are the main fields which are supported by bank. Payment Resources with any of the unsupported Enums will be Rejected.

LocalInstrument: Following Enums are supported by Bank

OB Enum	Mode of Payment
BH.OBF.DNS	Fawri
BH.OBF.NRT	Fawri+

DebtorAccount\SchemeName: Enum values for this fields depends on the LocalInstrument value as follows.

LocalInstrument	SchemeName
BH.OBF.DNS	BH.OBF.IBAN
BH.OBF.NRT	BH.OBF.IBAN

AppliedAuthenticationApproach: For this field only supported Enum is SCA as for each payment request must go through SCA.

3. If the payment Consent request is as per expectation then bank will return a Consent ID

Sample Request

POST <https://apis.meem.bh/open-banking/v3.1/pisp/domestic-payment-consents>

Mandatory field(s)

Authorization =Application_Access_Token

Body =

```
{
  "Data":{
    "ReadRefundAccount":"No",
    "Initiation":{
      "InstructionIdentification":"1618822238001147",
      "EndToEndIdentification":"FRESCO.21302.GFX.20",
      "LocalInstrument":"BH.OBF.DNS",
      "InstructedAmount":{
        "Amount":"3.000",
        "Currency":"BHD"
      },
      "DebtorAccount":{
        "SchemeName":"BH.OBF.BBAN",
        "Identification":"A123456B789"
      },
      "CreditorAccount":{
        "SchemeName":"BH.OBF.IBAN",
        "Identification":"BH21234567890000009",
        "Name":"Test User"
      },
      "RemittanceInformation":{
        "RemittanceDescription":"Internal ops code 123456"
        "Reference":"FRESCO-000001"
      }
    },
    "Authorisation":{
      "AuthorisationType":"Single",
      "CompletionDateTime":"2021-10-11T05:52:37.236Z"
    },
    "Risk":{
  }
```

```
}
```

Sample Response

```
{
```

```
  "Meta": {},
```

```
  "Risk": {},
```

```
  "Links": {
```

```
    "Self": "https://XYZURL.com"
```

```
  },
```

```
  "Data": {
```

```
    "Status": "AwaitingAuthorisation",
```

```
    "StatusUpdateDateTime": "2021-04-26T14:40:56Z",
```

```
    "CreationDateTime": "2021-04-26T14:40:56Z",
```

```
    "Authorisation": {
```

```
      "CompletionDateTime": "2021-10-11T05:52:37.236Z",
```

```
      "AuthorisationType": "Single"
```

```
    },
```

```
    "ConsentId": "4037837f-72d5-4ddf-9774-91aef31611cf",
```

```
    "ReadRefundAccount": "No",
```

```
    "Initiation": {
```

```
      "LocalInstrument": "BH.OBF.DNS",
```

```
      "DebtorAccount": {
```

```
        "SchemeName": "BH.OBF.BBAN",
```

```
        "Identification": "A12345B6789"
```

```
      },
```

```
      "RemittanceInformation": {
```

```
        "RemittanceDescription": "Internal ops code ABCD1234",
```

```
        "Reference": "FRESCO-10111"
```

```
      },
```

```
    "EndToEndIdentification": "FRESCO.21302.GFX.20",
```

```
    "InstructionIdentification": "1618822238001140",    "CreditorAccount": {
```

```
      "SchemeName": "BH.OBF.IBAN",
```

```
      "Identification": "BHABCDEF000A123456B780",
```

```
"Name": "TEST NAME"  
},  
"InstructedAmount": {  
  "Amount": "3.000",  
  "Currency": "BHD"  
}  
}  
}  
}
```

5.1.3 GET /domestic-payment-consents/{ConsentId}

A PISP can retrieve a payment consent resource that they have created to check its status with the consent id provided by bank.

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/pisp/domestic-payment-consents/{ConsentId}>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```
{
  "Meta": {},
  "Risk": {},
  "Links": {
    "Self": "https://XYZURL.com"
  },
  "Data": {
    "Status": "AwaitingAuthorisation",
    "StatusUpdateDateTime": "2021-04-26T14:40:56Z",
    "CreationDateTime": "2021-04-26T14:40:56Z",
    "Authorisation": {
      "CompletionDateTime": "2021-10-11T05:52:37.236Z",
      "AuthorisationType": "Single"
    },
    "ConsentId": "4037837f-72d5-4ddf-9774-91aef31611cf",
    "ReadRefundAccount": "No",
    "Initiation": {
      "LocalInstrument": "BH.OBF.DNS",
      "DebtorAccount": {
        "SchemeName": "BH.OBF.BBAN",
        "Identification": "A12345B6789"
      }
    }
  }
}
```

```

"RemittanceInformation": {
  "RemittanceDescription": "Internal ops code ABCD1234",
  "Reference": "FRESCO-10111"
},
"EndToEndIdentification": "FRESCO.21302.GFX.20",
"InstructionIdentification": "1618822238001140",
"CreditorAccount": {
  "SchemeName": "BH.OBF.IBAN",
  "Identification": "BHABCDEF000A123456B780",
  "Name": "TEST NAME"
},
"InstructedAmount": {
  "Amount": "3.000",
  "Currency": "BHD"
}
}
}
}
}

```

5.1.4 GET /domestic-payment-consents/{ConsentId}/funds-confirmation

The API endpoint allows the PISP to ask an ASPSP to confirm funds on a domestic-payment-consents resource and if the domestic-payment-consents resource has an Authorized status the bank will respond to a funds confirmation request with True/ False.

Sample Request

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/pisp/domestic-payment-consents/{ConsentId}/fundsconfirmation>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```
{  
  "Data": {  
    "FundsAvailableResult": {  
      "FundsAvailableDateTime": "2021-04-26T09:37:52.049+03:00",  
    }  
  }  
}
```

5.2 Domestic Payments Initiation

5.2.1 POST /domestic-payments

Once the domestic-payment-consents has been authorized by the user/customer, the PISP can proceed to submit the domestic-payment for processing

The PISP must ensure that the Initiation and Risk sections of the domestic-payment match the corresponding Initiation and Risk sections of the domestic-payment-consents resource. If the two do not match, the ASPSP must not process the request

If the Request is as per expectations then Bank will initiate the payment and share the DomesticPaymentId which may be used by PISP to get status of the payment request.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

POST <https://apis.meem.bh/open-banking/v3.1/pisp/domestic-payments>

Mandatory Fields:

Authorization =User_Access_Token

Body =

```
{  
  "Data":{
```

```
"ConsentId": "b5cbc07e-0cfe-453e-8c72-29cb158d7e4e",
"Initiation": {
  "InstructionIdentification": "1618822238001145",
  "EndToEndIdentification": "FRESCO.21302.GFX.20",
  "LocalInstrument": "BH.OBF.DNS",
  "InstructedAmount": {
    "Amount": "3.000",
    "Currency": "BHD"
  },
  "DebtorAccount": {
    "SchemeName": "BH.OBF.BBAN",
    "Identification": "A12345B6789"
  },
  "CreditorAccount": {
    "SchemeName": "BH.OBF.IBAN",
    "Identification": "BH12345678980120009",
    "Name": "Test User"
  },
  "RemittanceInformation": {
    "RemittanceDescription": "Internal ops code 123456",
    "Reference": "FRESCO-10111"
  }
},
"Risk": {
}
}
```

Sample Response

```
{
  "Data": {
```



```
"DomesticPaymentId": "3XXXXXXXXXXXXX01",
"ConsentId": "4037837f-72d5-4ddf-9774-91aef31611cf",
"CreationDateTime": "2021-04-26T17:53:12.122+03:00",
"Status": "AcceptedSettlementCompleted",
"StatusUpdateDateTime": "2021-04-26T17:53:12.122+03:00",
"Initiation": {
  "InstructionIdentification": 1618822238001140,
  "EndToEndIdentification": "FRESCO.21302.GFX.20",
  "LocalInstrument": "BH.OBF.DNS",
  "InstructedAmount": {
    "Amount": 3.000,
    "Currency": "BHD"
  },
  "Debtor": {
    "Name": null
  }
}
}
```

5.2.2 GET /domestic-payments/{DomesticPaymentId}

A PISP can retrieve the domestic-payment to check its status using DomesticPaymentId provided bank.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/pisp/international-paymentconsents/{domesticPaymentId}> **Mandatory Fields:**

Authorization =User_Access_Token domesticPaymentId = domesticPaymentId Received in

Response of Post Domestic-Payments [Sample Response](#)

```
{
  "Data": {
    "DomesticPaymentId": "3XXXXXXXXXX0501",
    "ConsentId": "123456",
    "CreationDateTime": "2021-04-26T17:54:36.55+03:00",
    "Status": "AcceptedSettlementCompleted",
    "StatusUpdateDateTime": "2021-04-26T17:54:36.55+03:00",
    "Charges": {
      "ChargeBearer": "Shared",
      "Amount": {
        "Amount": "0",
        "Currency": ""
      }
    },
  },
  "Initiation": {
    "InstructionIdentification": "1618822238001140",
    "EndToEndIdentification": "FRESCO.21302.GFX.20",
    "LocalInstrument": "Fawri",
    "InstructedAmount": {
      "Amount": "3.000",
      "Currency": "BHD"
    }
  },
  "DebtorAccount": {
    "Identification": "ABCD12345",
    "Name": ""
  },
  "CreditorAccount": {
    "Identification": "BH27GULF00039900020949",
    "Name": "Test User",
    "CreditorPostalAddress": {
      "AddressLine": "",
      "TownName": ""
    }
  }
}
```

```
"Country": ""
}
},
"RemittanceInformation": {
  "RemittanceDescription": "Internal ops code 123456",
  "Reference": "FRESCO-10111"
},
"Links": {
  "Self": "http://www.gib.com"
}
}
```

6.3 International Payments Consents

Every domestic Payment request must be preceded with Payment consents, and this Payment consent must be authorized by the customer at GIB Bahrain page.

5.3.1 POST /International-payment-consents

1. User should give consent to a PISP to make an international payment on behalf a customer, after this the PISP uses this API to create a payment-consent resource
2. The International payment consent resource includes all the required payment related fields which are consented by the user with PISP. Below are the main fields which are supported by bank. Payment Resources with any of the unsupported Enums will be Rejected.

LocalInstrument: For this Field Supported Enum is only BH.OBF.SWIFT

InstructionPriority: For this Field Supported Enum is only Normal

ChargeBearer: Following Enums are supported by bank

BorneByCreditor
BorneByDebtor
Shared

ExchangeRateInformation: International Transfers/Payments will always be processed with exchange rates by Bank. Values provided in this data set will be ignored.

DebtorAccount\SchemeName: For this Field Supported Enum is only BH.OBF.IBAN

CreditorAgent\ SchemeName: For this Field Supported Enum is only BH.OBF.BICFI

DestinationCountryCode field is mandatory if Creditor account scheme name is not IBAN

3. If the payment Consent request is as per expectation then bank will return a Consent ID

Sample Request

POST <https://apis.meem.bh/open-banking/v3.1/pisp/international-payment-consents>

Mandatory field(s)

Authorization =Application_Access_Token

Body =

```
{  
  "Data":{  
    "ReadRefundAccount":"No",  
    "Initiation":{
```

```
"InstructionIdentification": "161828132131",
"EndToEndIdentification": "20190616B0021",
"LocalInstrument": "BH.OBF.SWIFT",
"InstructionPriority": "Normal",
"CurrencyOfTransfer": "BHD",
"Purpose": "Home",
"ChargeBearer": "Shared",
"DestinationCountryCode": "KW",
"InstructedAmount": {
  "Amount": "7.00",
  "Currency": "BHD"
},
"CreditorAgent": {
  "Identification": "ABCD1234",
  "SchemeName": "BH.OBF.BICFI"
},
"CreditorAccount": {
  "SchemeName": "BH.OBF.BBAN",
  "Identification": "KW123456789012345",
  "Name": "Test User"
},
"DebtorAccount": {
  "SchemeName": "BH.OBF.BBAN",
  "Identification": "123456789",
  "Name": "Test User"
},
"RemittanceInformation": {
  "RemittanceDescription": "Internal ops code 123456",
  "Reference": "FRESCO-123456"
},
"Creditor": {
  "Name": "Test User",
  "PostalAddress": {
```

```
"AddressType": "Correspondence",
"Department": "department1",
"SubDepartment": "sub dept",
"StreetName": "XYZ Avenue",
"BuildingNumber": "27",
"PostCode": "GU31 2ZZ",
"TownName": "Sparsholt",
"CountrySubDivision": "Wessex",
"Country": "KW",
"AddressLine": [
  "Flat 7",
  "ABCD XYZ"
]
}
}
},
"Risk": {
}
}
```

Sample Response

```
{
  "Meta": {},
  "Risk": {},
  "Links": {
    "Self": "https://url.COM/"
  },
  "Data": {
    "Status": "AwaitingAuthorisation",
    "StatusUpdateDateTime": "2021-04-27T11:11:57Z",
    "CreationDateTime": "2021-04-27T11:11:57Z",
  }
}
```

```
"ConsentId": "7c05170e-537e-4c9d-ab15-9be6451ffa34",
"ReadRefundAccount": "No",
"Initiation": {
  "CreditorAgent": {
    "SchemeName": "BH.OBF.BICFI",
    "Identification": "ABCD12345"
  },
  "DebtorAccount": {
    "SchemeName": "BH.OBF.BBAN",
    "Identification": "A12345B678",
    "Name": "Mohan Didymus"
  },
  "EndToEndIdentification": "20190616B0021",
  "CurrencyOfTransfer": "BHD",
  "InstructionIdentification": "161828132131",
  "CreditorAccount": {
    "SchemeName": "BH.OBF.BBAN",
    "Identification": "KW12345678901234567",
    "Name": "Test User"
  },
  "Purpose": "Home",
  "ChargeBearer": "Shared",
  "InstructionPriority": "Normal",
  "LocalInstrument": "BH.OBF.SWIFT",
  "RemittanceInformation": {
    "RemittanceDescription": "Internal ops code 123456",
    "Reference": "FRESCO-101111"
  },
  "Creditor": {
    "PostalAddress": {
      "StreetName": "Test Name",
      "CountrySubDivision": "Wessex",
      "Department": "department1",
```

```

    "AddressLine": [
      "Flat 7",
      "ABCD XYZ"
    ],
    "BuildingNumber": "27",
    "TownName": "Sparsholt",
    "Country": "KW",
    "SubDepartment": "sub dept",
    "AddressType": "Correspondence",
    "PostCode": "GU31 2ZZ"
  },
  "Name": "TEST USER"
},
  "InstructedAmount": {
    "Amount": "7.00",
    "Currency": "BHD"
  }
}
}
}
}

```

5.3.2 GET /international-payment-consents/{ConsentId}

A PISP can retrieve a payment consent resource that they have created to check its status with the consent id provided by bank.

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/pisp/international-payment-consents/{ConsentId}>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```

{
  "Meta": {},

```



```
"Risk": {},
"Links": {
  "Self": "https://url.COM/"
},
"Data": {
  "Status": "AwaitingAuthorisation",
  "StatusUpdateDateTime": "2021-04-27T11:11:57Z",
  "CreationDateTime": "2021-04-27T11:11:57Z",
  "ConsentId": "7c05170e-537e-4c9d-ab15-9be6451ffa34",
  "ReadRefundAccount": "No",
  "Initiation": {
    "CreditorAgent": {
      "SchemeName": "BH.OBF.BICFI",
      "Identification": "ABCD12345"
    },
    "DebtorAccount": {
      "SchemeName": "BH.OBF.BBAN",
      "Identification": "A12345B678",
      "Name": "Mohan Didymus"
    },
    "EndToEndIdentification": "20190616B0021",
    "CurrencyOfTransfer": "BHD",
    "InstructionIdentification": "161828132131",
    "CreditorAccount": {
      "SchemeName": "BH.OBF.BBAN",
      "Identification": "KW12345678901234567",
      "Name": "Test User"
    },
    "Purpose": "Home",
    "ChargeBearer": "Shared",
    "InstructionPriority": "Normal",
    "LocalInstrument": "BH.OBF.SWIFT",
    "RemittanceInformation": {
```

```
"RemittanceDescription": "Internal ops code 123456",
"Reference": "FRESCO-101111"
},
"Creditor": {
  "PostalAddress": {
    "StreetName": "Test Name",
    "CountrySubDivision": "Wessex",
    "Department": "department1",
    "AddressLine": [
      "Flat 7",
      "ABCD XYZ"
    ],
    "BuildingNumber": "27",
    "TownName": "Sparsholt",
    "Country": "KW",
    "SubDepartment": "sub dept",
    "AddressType": "Correspondence",
    "PostCode": "GU31 2ZZ"
  },
  "Name": "TEST USER"
},
"InstructedAmount": {
  "Amount": "7.00",
  "Currency": "BHD"
}
}
}
```

5.3.3 GET /international-payment-consents/{ConsentId}/funds-confirmation

The API endpoint allows the PISP to ask an ASPSP to confirm funds on an International-paymentconsents resource and if the international-payment-consents resource has an Authorized status the bank will respond to a funds confirmation request with True/ False.

Sample Request

GET <https://apis.meem.bh/open-banking/v3.1/pisp/international-payment-consents/{ConsentId}/fundsconfirmation>

Mandatory field(s)

Authorization =Application_Access_Token

ConsentID = ConsentId

Sample Response

```
{
  "Data": {
    "FundsAvailableResult": {
      "FundsAvailableDateTime": "2021-04-26T09:37:52.049+03:00",
      "FundsAvailable": true
    }
  }
}
```

5.4 International Payments Initiation

5.4.1 POST /international-payments

Once the international-payment-consents has been authorized by the user/customer, the PISP can proceed to submit the international -payment for processing

The PISP must ensure that the Initiation and Risk sections of the international -payment match the corresponding Initiation and Risk sections of the international -payment-consents resource. If the two do not match, the ASPSP will not process the request

If the Request is as per expectations then Bank will initiate the payment and share the internationalpaymentId which may be used by PISP to get status of the payment request.

Security Requirements:

- Mutual TLS

- Digital Signature
- TPP Application Access Token

Sample Request

POST <https://apis.meem.bh/open-banking/v3.1/pisp/international-payments> **Mandatory**

Fields:

Authorization =User_Access_Token

Body =

```
{
  "Data":{
    "ReadRefundAccount":"No",
    "Initiation":{
      "InstructionIdentification":"160001471",
      "EndToEndIdentification":"20190616B0021",
      "LocalInstrument":"BH.OBF.SWIFT",
      "InstructionPriority":"Normal",
      "CurrencyOfTransfer":"KWD",
      "Purpose":"Home",
      "ChargeBearer":"Shared",
      "DestinationCountryCode":"KW",
      "InstructedAmount":{
        "Amount":"7.00",
        "Currency":"KWD"
      },
    },
    "CreditorAgent":{
      "Identification":"NABCDEDED",
      "SchemeName":"BH.OBF.BICFI"
    },
    "CreditorAccount":{
      "SchemeName":"BH.OBF.BBAN",
      "Identification":"BH1234567890000001",
      "Name":"ACME Inc"
    },
  },
}
```

```
"DebtorAccount":{
  "SchemeName":"BH.OBF.BBAN",
  "Identification":"3A12345B6789",
  "Name":"Test User"
},
"RemittanceInformation":{
  "RemittanceDescription":"Internal ops code 5120101",
  "Reference":"FRESCO-101111"
},
"Creditor":{
  "Name":"ACME Inc",
  "PostalAddress":{
    "AddressType":"Address Line 1",
    "Department":"department1",
    "SubDepartment":"sub dept",
    "StreetName":"Acacia Avenue",
    "BuildingNumber":"27",
    "PostCode":"GU31 2ZZ",
    "TownName":"Sparsholt",
    "CountrySubDivision":"Wessex",
    "Country":"KW",
    "AddressLine":[
      "Flat 7",
      "Acacia Lodge"
    ]
  }
}
},
"Risk":{
}
}
```

Sample Response

```
{  
  "Data": {  
    "InternationalPaymentId": "123ABCD12345210990005",  
    "ConsentId": 123456,  
    "CreationDateTime": "2021-04-09T12:37:01.808+03:00",  
    "Status": "AcceptedCreditSettlementCompleted",  
    "StatusUpdateDateTime": "2021-04-09T12:37:01.808+03:00",  
    "Initiation": {  
      "InstructionIdentification": 12700088912885,  
      "EndToEndIdentification": "20190616B002",  
      "Purpose": null,  
      "ExtendedPurpose": null,  
      "ChargeBearer": "Sender",  
      "CurrencyOfTransfer": "KWD",  
      "DestinationCountryCode": null,  
      "InstructedAmount": {  
        "Amount": 10.00,  
        "Currency": "BHD"  
      },  
      "DebtorAccount": {  
        "SchemeName": "BH.OBF.IBAN",  
        "Identification": 12345678  
      },  
      "Creditor": {
```

```
"Name": "test user",

"PostalAddress": {

  "AddressLine": "XYZBank, Diplomatic Area",

  "TownName": "salmiya",

  "Country": "BH"

}

},

"CreditorAgent": {

  "Identification": "A123456789",

  "Name": null,

  "PostalAddress": {

    "AddressLine": "XYZBank, Diplomatic Area",

    "Country": null

  }

},

"RemittanceInformation": {      "RemittanceDescription": "Personal-Education",

  "Reference": "Education"

}

}

}

}
```

5.4.2 GET / international-payments/{ internationalPaymentId}

A PISP can retrieve the international-payment to check its status using internationalPaymentId provided bank.

Security Requirements:

- Mutual TLS
- Digital Signature
- TPP Application Access Token

Sample Request

GET [https://apis.meem.bh/open-banking/v3.1/pisp/international-payments /{InternationalPaymentId}](https://apis.meem.bh/open-banking/v3.1/pisp/international-payments/{InternationalPaymentId})

Mandatory Fields:

Authorization =User_Access_Token

InternationalPaymentId = InternationalPaymentId Received in Response of Post International-Payments

Sample Response

```
{  
  "Data": {  
    "InternationalPaymentId": "123ABCD12345210990005",  
    "ConsentId": 123456,  
    "CreationDateTime": "2021-04-09T12:37:01.808+03:00",  
    "Status": "AcceptedCreditSettlementCompleted",  
    "StatusUpdateDateTime": "2021-04-09T12:37:01.808+03:00",  
    "Initiation": {  
      "InstructionIdentification": 12700088912885,  
      "EndToEndIdentification": "20190616B002",  
      "Purpose": null,  
      "ExtendedPurpose": null,  
      "ChargeBearer": "Sender",  
      "CurrencyOfTransfer": "KWD",  
      "DestinationCountryCode": null,  
      "InstructedAmount": {  
        "Amount": 10.00,  
        "Currency": "BHD"  
      }  
    }  
  }  
}
```



```
},
  "DebtorAccount": {
    "SchemeName": "BH.OBF.IBAN",
    "Identification": 12345678
  },
  "Creditor": {
    "Name": "test user",
    "PostalAddress": {
      "AddressLine": "XYZBank, Diplomatic Area",
      "TownName": "salmiya",
      "Country": "BH"
    }
  },
  "CreditorAgent": {
    "Identification": "A123456789",
    "Name": null,
    "PostalAddress": {
      "AddressLine": "XYZBank, Diplomatic Area",
      "Country": null
    }
  },
  "RemittanceInformation": {
    "RemittanceDescription": "Personal-Education",
    "Reference": "Education"
  }
}
}
```

7. PIS Swagger File

Please Find the swagger file for PIS



payment-swagger-3
.1.6.yaml