

How to Smart Accounts



Author: Gabriela Guță

Reviewed by: Bogdan-Robert Guriță

Version: 2.2

Date: 27.07.2023

UPDATES HISTORY				
Version	Date	Changes details	Author	Pages
1.0	13.12.2022	Initial version	Gabriela Guță	113
2.0	31.03.2023	<ul style="list-style-type: none"> • Added flows for the following banks: First Bank, UniCredit Bank, Revolut, OTP Bank, Garanti Bank, Libra (only mentioned that it is available in Prod environment); • Exemplified flow for initConsent in Sandbox and Prod environments; • Added consents' granularity per integrated bank; • Added multiple consents' handling per integrated bank; • Added pagination vs bulk transactions per integrated bank. 	Bogdan-Robert Guriță	187
2.1	03.05.2023	<ul style="list-style-type: none"> • Added AIS flow for Libra • Added mentions regarding error flow • PSU-ID-TYPE (single value: "ALL") for UniCredit will be sent automatically by SACC • Update GET Banks API • Modified Revolut prod Flow • Mentioned parameter "next" • Added error mapping and handling 	Bogdan-Robert Guriță	202
2.2	27.07.2023	<ul style="list-style-type: none"> • Added mention about PSU-ID-TYPE for Raiffeisen Bank 	Bogdan-Robert Guriță	202

Contents

1.	Scope.....	4
2.	Glossary.....	5
3.	Postman Setup.....	7
4.	Integrated banks	10
5.	Account Information Consent flow.....	11
1.	Partner's authentication	11
2.	Get Banks	11
3.	Check email.....	11
4.	initConsent.....	11
5.	Get Consent	15
6.	Get Consent Status	18
7.	Get Accounts.....	19
8.	Get Balances.....	21
9.	Get Transactions	22
10.	GetTransactionDetails.....	24
11.	GetAccountDetails	25
12.	DeleteConsent.....	26
6.	Banks flow	28
1.	Banca Transilvania - Flow.....	28
2.	ING Bank – Flow	43
3.	Banca Comerciala Romana - Flow.....	55
4.	BRD PJ (corporate) - Flow	67
5.	BRD PF (individuals) - Flow.....	78
6.	Raiffeisen Bank - Flow	88
7.	CEC Bank - Flow.....	100
8.	First Bank - Flow.....	116
9.	UniCredit Bank - Flow	129
10.	Revolut - Flow	143
11.	OTP Bank - Flow	159
12.	Garanti BBVA – Flow	172
13.	Libra – Flow	185
7.	Generic APIs (available in all banks' flows)	197
1.	Banks	197
2.	CheckEmail.....	200
8.	Error mapping & handling linked to a consent's status	202

1. Scope

This document comes in addition to “Smart Accounts - Technical manual for intermediary_v1” for details regarding especially the functional flow. This document is addressed to Smart FinTech's and Smart Accounts' partners for better understanding over the AIS flow implemented and each bank particularities.

Please note that the document has been created by Smart FinTech and it cannot be disclosed or redistributed.

2. Glossary

PSD2 = Payment Service Directive 2, the revised version of the EU directive that was implemented in 2007

TPP = Third-Party Provider

Intermediary/Partner = The legal entity that integrates with Smart Accounts in order to access its client's account information, with their consent and in their benefit

X.509 = standard format for public key certificates, digital documents, that securely associate cryptographic key pairs with identities such as organisations, websites or individuals. In order to integrate with Smart Accounts, a TLS server certificate is needed (QWAC-Qualified Website Authentication Certificate)

PSU (Payment Services User) = the account's holder that gives his permission to account's information

AISP (Account Information Service Provider) = the third-party provider who collects an account holder's information. Smart FinTech is an AISP and also TPP

AIS (Account Information Service) = the service that provides the PSU a consolidated view of his finances: bank balance, transaction history from multiple bank accounts

API (Application Programming Interface) = Open API is a free-to-use, publicly available APIs (the ones shared by each bank on their developer portal site)

EBA (European Banking Authority) = the authority that ensures consistency in regulation and supervision across the European banking sector. For PSD2, they are responsible for issuing Regulatory Technical Standards and guidelines

SCA (Strong Customer Authentication) prescribes that authentication should occur using a combination of two factors from the categories:

- Knowledge – "something you know" (password, PIN, etc.)
- Possession – "something you have" (token, smartphone, application)
- Inherence – "something you are" (fingerprint, voice recognition)

eIDAS (Electronic Identification, Authentication and Trust Services) = a framework that sets standards for electronic identification and trust services. For PSD2 it's about SCA standards

SaaS (Software-as-a-service) = a type of software delivered online that enables users and companies to gain access to applications without the need to install or maintain software locally

GDPR (General Data Protection Regulation) = a regulation by which the European Parliament, the European Council and the European Commission intend to strengthen and unify data protection for individuals within the European Union

Open Banking ecosystem = refers to all elements that facilitate the operation of Open Banking (enabled by PSD2): API standards, the governance, systems, processes, security and procedures used to support participants

Decoupled (flow) authentication = the authentication of the PSU occurs outside of the interaction with the intermediary page (the PSU must log in in the internet banking application to authorise the consent)

Redirect (flow) authentication = the authentication of the PSU occurs through a redirect URL in the bank's page where he must complete his internet banking's credential to log in and authorise the consent

Resource = term used to also describe the consent

PF = personal account (retail)

PJ = corporate account (corporate/business)

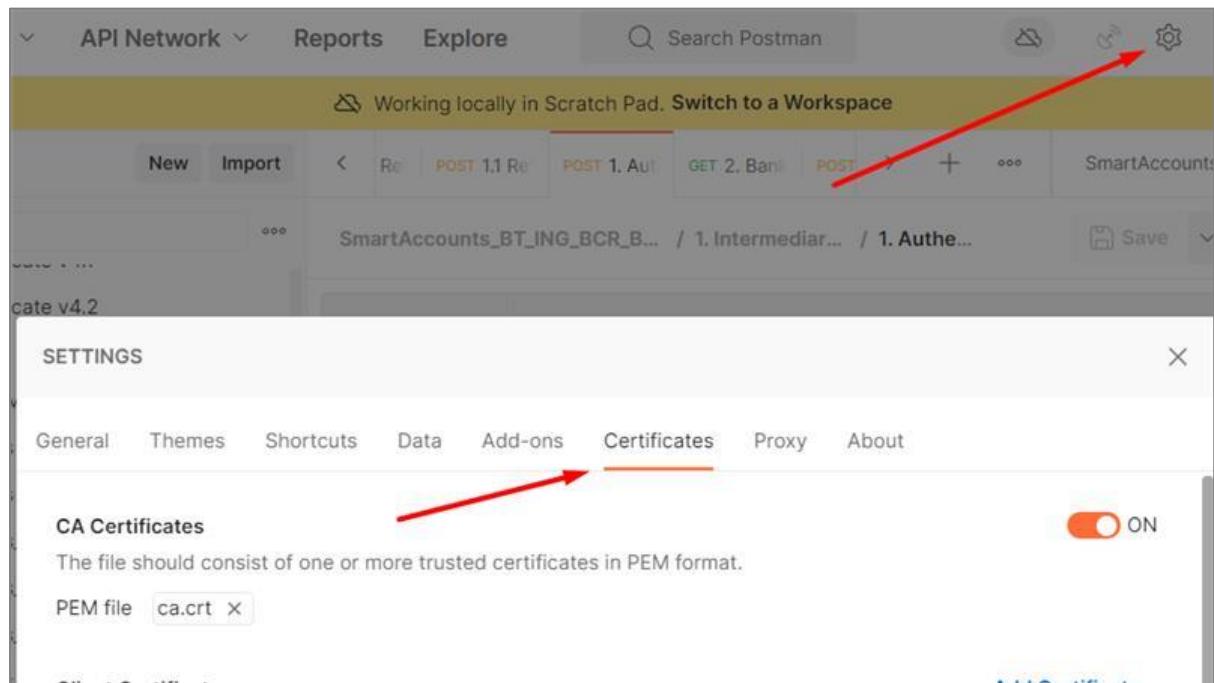
Sandbox (environment) = isolated copy of production environment designed for developers to try out APIs. Mimics all interactions with banks' APIs. It returns test (mock) data.

Production (environment) = environment used to return live data

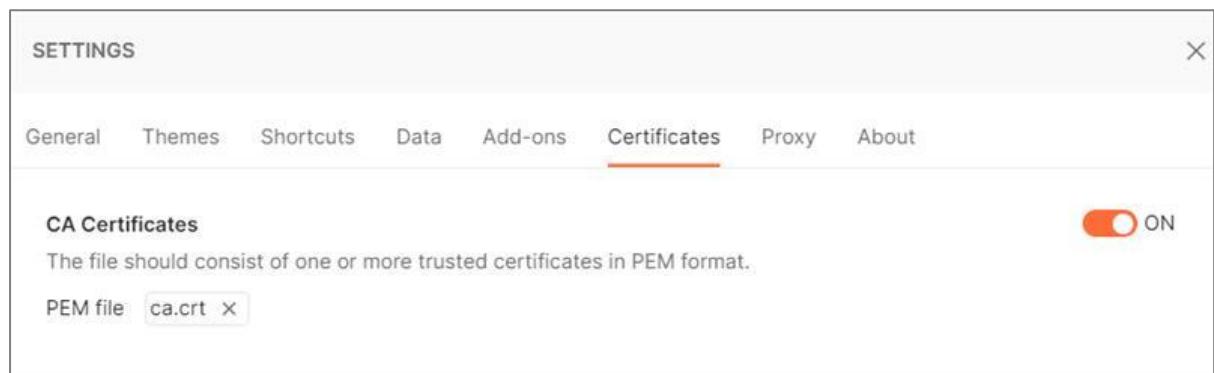
3. Postman Setup

The first step in setting up Smart Accounts is to add the related collections, certificates and environments to Postman. For the correct addition, the steps are shown graphically:

1. In the Settings menu, go to the "Certificates" tab



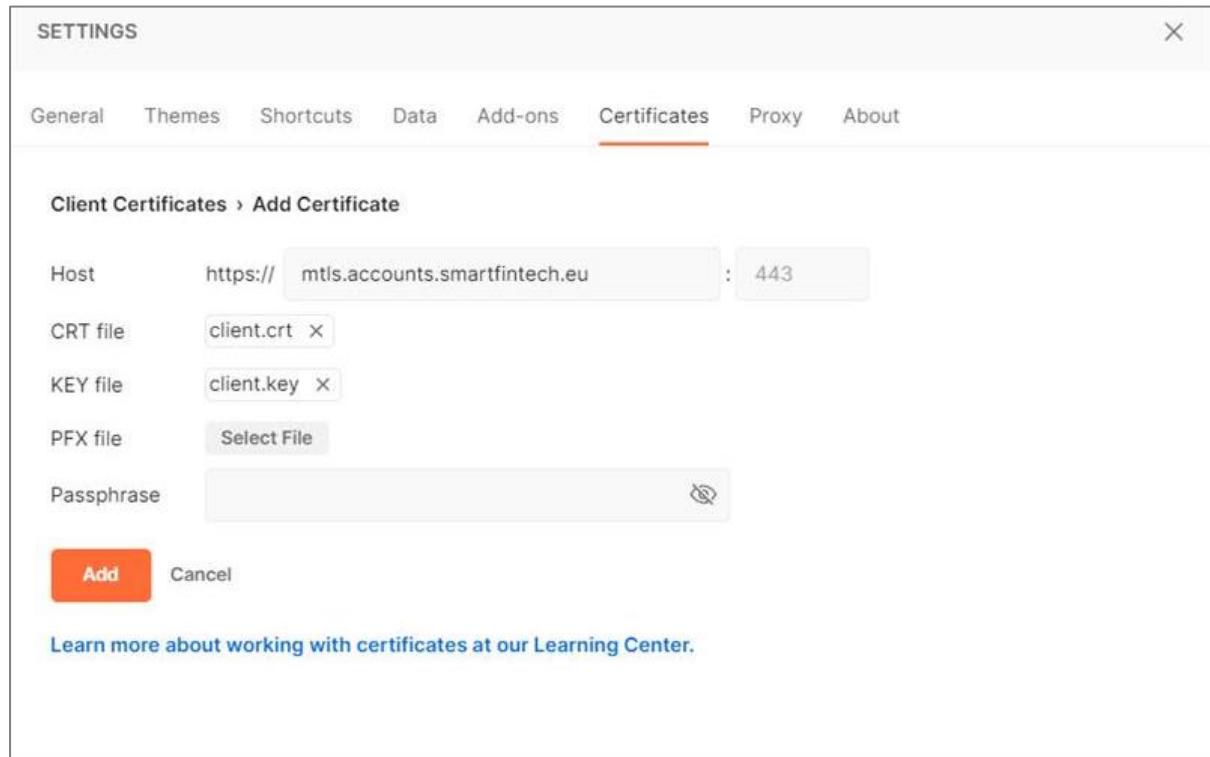
2. Add the file named "ca.crt" into CA Certificates field.



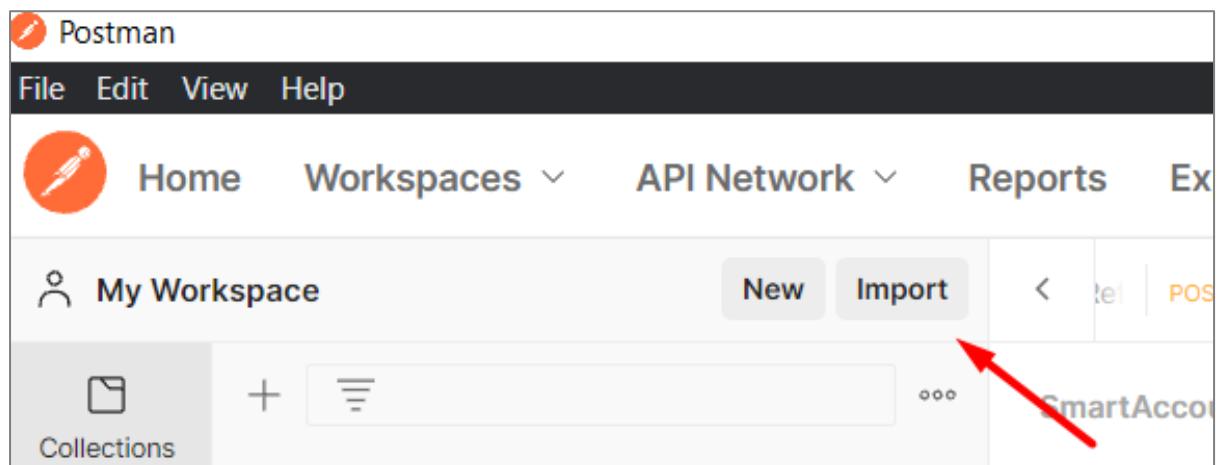
3. Press "Add Certificate", then:

- add the file "client.crt" into "CRT file" field

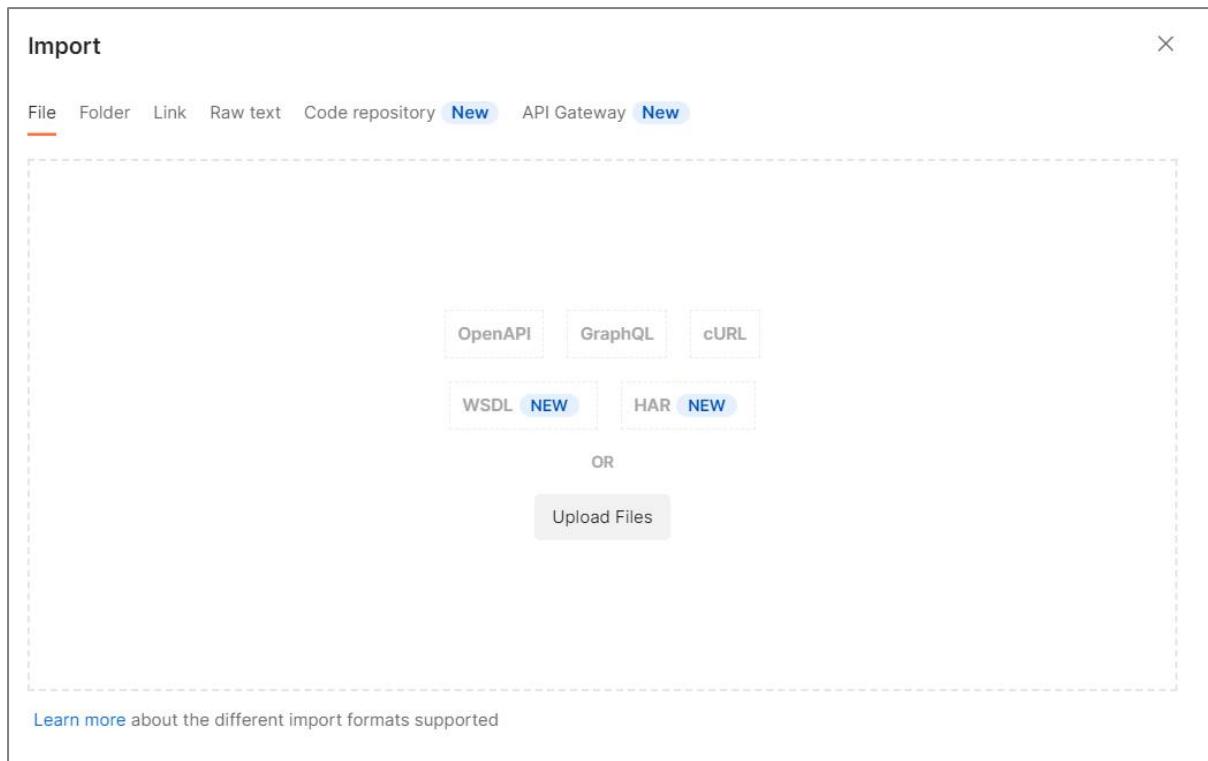
- add the file "client.key" into "Key File" field



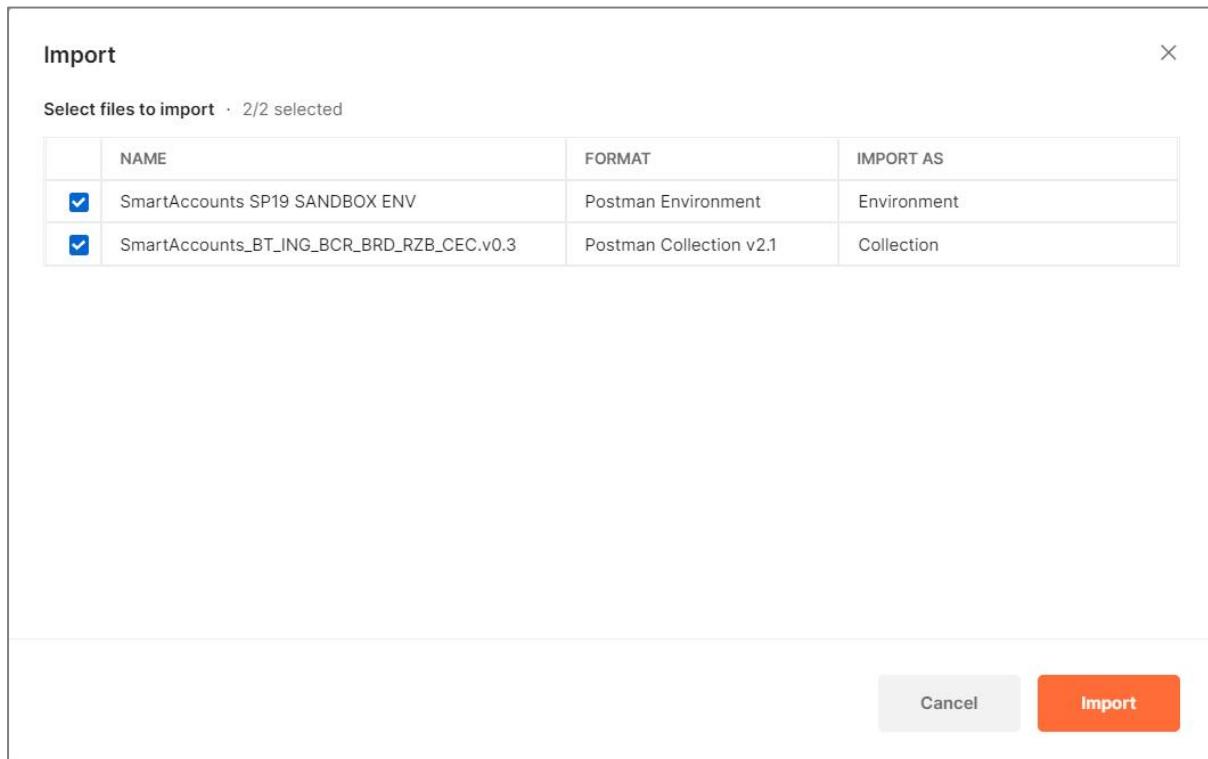
4. Press "Import" in order to add the collection and the environment.



You can drag and drop, or search for the exact path by clicking "Upload Files".



After the environment and collection have been successfully added, click "Import".



4. Integrated banks

The list of integrated banks with Smart Accounts (SA):

1. Banca Transilvania (BT) – redirect flow
2. Banca Comerciala Romana (BCR) – redirect flow
3. ING Bank (ING) – redirect flow
4. BRD Groupe Societe Generale (BRD) – decoupled flow
5. Raiffeisen Bank (RZB) – redirect flow
6. CEC Bank (CEC) – redirect flow
7. First Bank (FB) – redirect flow
8. UniCredit Bank (UNC) – redirect flow
9. Revolut (REV) – redirect flow
10. OTP Bank (OTP) – redirect flow
11. Garanti BBVA (GRT) – redirect flow
12. Libra (LIB) – redirect flow

1. Redirect flow - the user will be redirected in a webpage (from bank) to authorise the consent.

2. Decoupled flow - after "initConsent", the user must independently login and authorise the consent in the internet banking application.

2.1. For BRD with "PSU-ID-Type" = "PF" (personal accounts) the user will have a push notification and must use the code provided via Consent initialisation confirmation mail.

2.2. For BRD with "PSU-ID-Type" = "PJ" (corporate accounts), the user does not have any notification, he must login in the internet banking application and authorise the consent. No code is needed for PJ.

5. Account Information Consent flow

1. Partner's authentication (API: authenticate/rest/api/token)

Creating a new consent flow will always start with partner's authentication. The partner must successfully authenticate to obtain a pair of access and refresh tokens and the ID of the consent that has been initiated. The pair of access and refresh token must be saved and used for calling the APIs relative to the generated consent ID. **It cannot be used to access information about other consents.**

The access token expires after 5 minutes, so it must be renewed using the refresh token provided ("POST refreshToken API"). The refresh token expires after 90 days.

When the "refreshToken" API will be called, a new pair of access and refresh tokens will be generated (API: /authenticate/rest/api/refreshToken).

In order to access the account's information, the new pair of access and refresh tokens must be used. The previous "refreshToken" is invalidated and it cannot be used anymore.

In case the "client_ID" (generated and provided by Smart FinTech) is missing or the x.509 certificate is missing, the authenticate step cannot be finished and an error will be raised. If the certificate has expired or the "client_ID" is not the correct one, an "Unauthorised" error will be raised. For the rest of technical errors, please check the technical document.

2. Get Banks (API: /core/rest/api/banks)

"GET Banks" returns the list of integrated and available banks. Every bank object contains the object "requiredParamters" which includes the parameters "requiresPSUIdType", "requiresPSUId" and "requiresiban". The parameters are boolean type and the value "true" means that the respective parameter should be sent on request.

All API calls must be made using the "codeBank" specific for each bank, otherwise an error will be raised.

3. Check email (API: core/rest/api/checkPsuEmail)

"CheckEmail" API will check the PSU's email address in Smart Accounts' database. If the email address is already added, then this means that the PSU already has a Smart Accounts account. For this case, the user must be informed that he must accept the Smart Accounts Terms and Conditions.

If the user is not found in Smart Accounts' database, then the user will be informed that there will be an account created in the aforementioned database and he must agree with the Smart Accounts Terms and Conditions. The API's purpose is only to validate if the user must be informed about the Smart Accounts account's creation.

4. initConsent - *Consent initialisation* (core/rest/api/initConsent/{{internalConsentId}})

At this point, the consent can be prepared to be sent to bank for authorisation. In order to initialise a consent, Smart Accounts needs the following information:

1. API body:

- **codeBank** – For which bank the PSU wants to create a consent
 - **periodOfValid** – the period is with default value = 90 days. If Smart Accounts receives a different value on this parameter, a "Bad request" error will be raised. After 90 days, the PSU must re-authorise in the bank's page. Please be aware that the re-authorisation in the bank's page will generate a new consent ID so you must keep the evidence of the PSU's consents to match the account information
 - **psuEmail** – it is the e-mail address of the PSU this address and it will be used to inform the user about the consent initialisation
 - **psuIntermediaryId** – is the unique ID associated to the PSU by the TPP
 - **redirectURL** – is the page where the PSU will be redirected after he finishes the authorisation phase in the bank's page. It is recommended to be redirect in the partner's page
 - **TCAccepted** – must be true, otherwise a "Bad request" error will be raised. It is partner's choice if for each new consent the user must check and agree to the Smart Accounts Terms and Conditions, or it can implement a different way of managing the PSU's T&C agreement

All the above parameters are mandatory for all banks!

Additionally, for RZB and OTP you need to add IBAN and currency on request. These banks have another parameter mandatory:

- **allAccounts** – with attributes "iban" and "currency"

Example of API body without parameter "allAccounts"

```
POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}
Send

Params Authorization • Headers (14) Body • Pre-request Script Tests Settings Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON ▾ Beautify

1
2     "codeBank": "BRD",
3     "periodOfValid": 90,
4     "psuEmail": "aaa.aaaa@mail.com",
5     "psuIntermediateId": "123",
6     "redirectURL": "https://google.com",
7     "TCaccepted": true
8
```

Example of API body with parameter "allAccounts" (RZB and OTP)

```
POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}
Params Authorization • Headers (11) Body • Pre-request Script Tests Settings Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON ▾
1 {
2   ...
3   "allAccounts": [
4     ...
5       {
6         ...
7           "iban": "R003RZBR0000069999999999",
8           ...
9           "currency": "RON"
10          ...
11        ],
12        ...
13        "codeBank": "RZB",
14        ...
15        "periodOfValidity": 90,
16        ...
17        "psuEmail": "aaa.aaaa@mail.com",
18        ...
19        "psuIntermediaryId": "123",
20        ...
21        "redirectURL": "https://google.com",
22        ...
23        "TCAccepted": true
24    }

```

2. **API header:**

- **Content-Type** – mandatory for all banks
- **PSU-IP-Address** – is **mandatory and without this information, there is a limitation over the number of API calls that we can make (4 calls maximum per day)**
- **PSU-ID** – it is generated by the bank. Mandatory for BRD, RZB, OTP
- **PSU-ID-Type** – mandatory for:
 - BRD - accepted values are: "PF", "PJ" or "PFA"
 - RZB – accepted value: "PJ" – **only for corporate accounts. Do not send PSU-ID-TYPE if you want to initiate a consent for an individual person.**

Consent granularity

For the following banks, the user can select the list of accounts to grant access for in the bank's page:

- BT
- ING
- CEC
- First Bank
- UNC
- REV
- LIB

For the following banks, the user can grant a more granular consent:

- CEC – the user can choose to grant consent for: accounts and/or balances and/or transactions
- FB – the user can choose to grant consent for: accounts and/or balances and/or transactions
- UNC – the user can choose to grant consent for: account details and/or balances and/or transactions

For the following banks, the user does not have the option of selecting from a list the accounts to grant access for in the bank's page:

- BCR
- BRD
- GRT (only login and confirm login from mobile app)

For the following banks, the IBAN will be sent on "initConsent":

- RZB

- OTP

BCR does not have the concept of "consent". If you will check the BCR's developer portal, you will see that there is no consent API defined. BCR considered that the authorisation in the bank's page is enough for gaining access to PSU's account information for 90 days. Smart FinTech has built an "initConsent" and "GetConsent" API to normalise the process for all banks.

ING has the same behaviour as BCR.

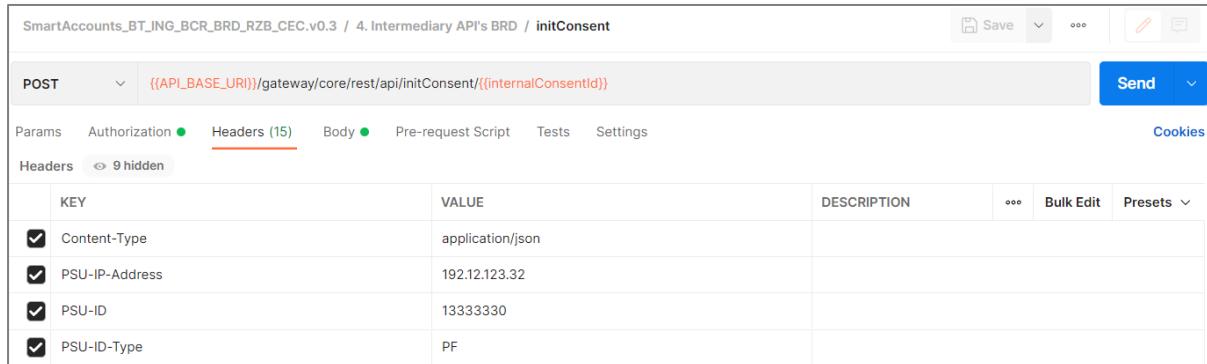
Smart Accounts has created internally the resources "GetConsent" and "GetConsentStatus".

BRD has a decoupled flow - this means that there will be no scaOAuth link returned. For BRD with "PSU-ID-Type" = "PF" (personal account) it will return a code on "consentID" parameter. That code must be used by the PSU to authorise the consent in the bank's internet banking application. After the PSU creates a new consent, he will receive an email from Smart Accounts (the consent initialisation email) and it will also be mentioned the code that he needs to introduce in order to authorise the consent.

We kindly suggest that you let the PSU know beforehand that BRD has a decoupled flow and he must authorise the consent by logging in into the internet banking's page. For BRD PF you must let the PSU know that he needs the code to authorise the consent.

If the "PSU-ID-Type" = "PJ" or "PFA" then the "consentID" will return null/empty. The PJ or PFA have the PSD2 flow implemented in another bank application that doesn't require any one-time password or code. The user is the one that should know which application he uses for that account.

initConsent request example with BRD decoupled:



The screenshot shows the Postman interface with the following details:

- Request Method: POST
- URL: `((API_BASE_URI))/gateway/core/rest/api/initConsent/{internalConsentId}`
- Headers tab selected, showing 15 headers:

 - Content-Type**: application/json
 - PSU-IP-Address**: 192.168.123.32
 - PSU-ID**: 13333330
 - PSU-ID-Type**: PF

initConsent response example with BRD decoupled:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "consentStatus": "RECEIVED", (please check GetConsentStatus API for details regarding consent's  
        statuses)  
        "consentId": "8000000022", (the code needed by user if the "codeBank" is BRD and PSU-ID-Type is  
        "PF". If the "codeBank" is different from BRD or the "codeBank" is BRD and PSU-  
        ID-Type is PJ or PFA then this parameter will be null/empty)  
        "_links": {  
            "scaOAuth": {  
                "href": null (for BRD will be null. If "codeBank" is different from BRD, then here will be sent  
                the code needed by user if the "codeBank" is BRD and PSU-ID-Type is PJ or PFA)  
            }  
        }  
    }  
}
```

the link for consent authorisation of the consent)

}

}

}

}

Creating multiple consents for the same user profile

Regarding **Banca Transilvania**, there is allowed a maximum of one consent per PSU and per bank's profile. If the user has a personal account and two business accounts, he will have 3 profiles: one per each business account and one per personal account. For this case, the user can have maximum 3 consents, one per each profile. In the bank's page, the PSU will have an additional screen where he can choose the profile and after that continue with the authorisation process. If the user will create a new consent for a profile that already has a valid consent, then the previous one will be invalidated by the bank and the user will have access only for the most recent consent created.

Regarding **BRD**, if the user is trying to initiate a consent for a profile that already has a valid consent, he will receive an error. The user must first delete the consent from the internet banking application and then he will be allowed to create another consent.

Creating multiple consents for the same PSU and the same bank account

The following banks allow maximum one consent per PSU per bank. If the user gives consent for an account which already has a valid consent, then the bank will invalidate the older consent and the user will have access only for the most recent consent created:

- BT
 - RZB
 - CEC
 - UNC
 - LIB
 - OTP

For the following banks, the user can have multiple valid consents at the same time. We kindly suggest letting the user know (or even restrict) that is not recommendable to have multiple consents at the same time.

- ING
 - BCR
 - FB
 - REV
 - GRT

5. Get Consent (/core/rest/api/consent/{{bank_name}}/{{internalConsentId}})

The purpose of this API is to check the consent details and the list of available accounts. The list of available accounts means the list of accounts for which the user has granted access through the authorised consent.

- For the following banks, the API response **will return the list of the selected accounts** by PSU and the parameter "availableAccounts" will be null:

- BT
- ING
- BRD (when PSU-ID-Type = PJ/PFA)
- RZB
- FB
- UNC
- OTP
- LIB

GetConsent response example:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO00BACX00000000000000000000"
          "currency": "RON" (only for RZB and LIB)
        }
      ],
      "balances": [
        {
          "iban": "RO00BACX00000000000000000000"
          "currency": "RON" (only for RZB and LIB)
        }
      ],
      "transactions": [
        {
          "iban": "RO00BACX00000000000000000000"
          "currency": "RON" (only for RZB and LIB)
        }
      ],
      "availableAccounts": null
    },
    "consentStatus": "valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-02",
    "validUntil": "2023-05-02",
    "recurringIndicator": true, (if "false", it means the respective consent is a one-time consent so it
                                can be used to call the API's resource only once). For the moment, SmartAccounts
                                makes the initConsent calls with default value set as "true" (required by bank)
    "combinedServiceIndicator": null, (only LIB implemented – flag present in Account Information
                                Consent Request which allows the use of consentId in the subsequent Payment
                                Initiation Request. It is relevant where account information services are needed
                                within a payment initiation)
    "creationDate": null
  }
}
```

2. For the following banks, the API response **will NOT return the list of the selected accounts** by PSU and the **parameter "availableAccounts" will be null**:
- CEC – if only an account is selected, then its IBAN will appear in "accounts", "balances" and "transactions"
 - REV

GetConsent response example:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [],
      "balances": [],
      "transactions": [],
      "availableAccounts": null
    },
    "consentStatus": "valid",
    "frequencyPerDay": null,
    "lastActionDate": "2023-02-08",
    "validUntil": "2023-05-08",
    "recurringIndicator": null,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

3. For the following banks, the API response **will NOT return the list of the selected accounts** by PSU and the **parameter "availableAccounts" will have the value "allAccounts"**:
- BCR
 - BRD (PSU-ID-Type = PF)
 - GRT

GetConsent response example:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [],
      "balances": [],
      "transactions": [],
      "availableAccounts": "allAccounts"
    },
    "consentStatus": "Valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-02",
    "validUntil": "2023-05-02",
    "recurringIndicator": true,
  }
}
```

```

        "combinedServiceIndicator": null,
        "creationDate": null
    }
}

```

6. Get Consent Status

(core/rest/api/consent/{{bank_name}}/{{internalConsentId}}/status)

This API will return the status of the consent sent on request. The status can also be checked on Get Consent API.

The list of possible statuses, according to Berlin Group Implementation Guidelines:

Status name	Description
received	The consent data have been received and are technically correct. The data is not authorised yet.
rejected	The consent data have been rejected e.g. since no successful authorisation has taken place.
partiallyAuthorised	The consent is due to a multi-level authorisation, some but not all mandated authorisations have been performed yet.
valid	The consent is accepted and valid for GET account data calls and others as specified in the consent object.
revokedByPsu	The consent has been revoked by the PSU towards the ASPSP.
expired	The consent expired.
terminatedByTpp	The corresponding TPP has terminated the consent by applying the DELETE method to the consent resource.

*The bold statuses are the ones we have encountered in production environment tests, until now.

GetConsentStatus response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "Valid"
    }
}

```

The bank's name must be according to the "codeBanks" from Get Banks, otherwise an error will be raised. **Validation performed on all resource APIs: If the bank's name and "internalConsentId" are not the same as the ones from "initConsent", an error will be raised.**

7. Get Accounts (core/rest/api/accounts/{{bank_name}}/{{internalConsentId}})

Returns a list of payment accounts to which an account access has been granted through the "initConsent" endpoint by the PSU. Parameter "PSU-IP-Address" is mandatory on API headers.

Example of GetAccounts headers

The screenshot shows the Postman interface with the 'Headers' tab selected. There is one header entry: 'PSU-IP-Address' with the value '192.12.123.32'. Other tabs like 'Params', 'Authorization', 'Body', etc., are visible at the top.

GetAccounts response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": [  
    {  
      "resourceId": "K13RONCRT0060214301", (resourceID is needed to call the next resource  
      APIs)  
      "iban": "RO98BTRLRONCRT0ABCDEFGHI",  
      "currency": "RON",  
      "name": "Cont de disponibil", (name of the account. For ING, REV and LIB it will be the  
      same as owner's name. Detailed at each bank)  
      "product": "Cont de disponibil", (Product name of the bank for this account, proprietary  
      definition)  
      "cashAccountType": "CurrentAccount", (ExternalCashAccountType1Code from ISO 20022)  
      "status": null, (not available for all banks. Represents account's status – can be "enabled",  
      "deleted, "blocked")  
      "bic": null,  
      "accountOfficerDescription": null, (only for LIB)  
      "openingDate": null, (only for LIB)  
      "customerSector": null, (only for LIB)  
      "customerType": null, (only for LIB)  
      "accountUsage": null, (only for LIB)  
      "workingBalance": null, (only for LIB)  
      "onlineActualBalance": null, (only for LIB)  
      "balances": [  
        {  
          "balanceAmount": {  
            "currency": "RON",  
            "amount": 485.55  
          },  
          "balanceType": "expected", (described after the response example)  
          "referenceDate": null, (indicates the date of the balance)  
          "creditLimitIncluded": "false", ( indicates if the credit limit of corresponding  
          account is included in the balance)  
        }  
      ]  
    }  
  ]  
}
```

```

        account is included in the calculation of the balance, were applicable)
        "creditDebitIndicator": null
    },
    {
        "balanceAmount": {
            "currency": "RON",
            "amount": 485.55
        },
        "balanceType": "expected",
        "referenceDate": null,
        "creditLimitIncluded": "true",
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
}
]
}

```

"balanceType" – The following balance types are defined:

1. **"closingBooked"**: Balance of the account at the end of the pre-agreed account reporting period. It is the sum of the opening booked balance at the beginning of the period and all entries booked to the account during the pre-agreed account reporting period.

For card-accounts, this is composed of

- invoiced, but not yet paid entries

2. **"expected"**: Balance composed of booked entries and pending items known at the time of calculation, which projects the end of day balance if everything is booked on the account and no other entry is posted.

For card accounts, this is composed of

- invoiced, but not yet paid entries,
- not yet invoiced but already booked entries and
- pending items (not yet booked)

3. **"authorised"**: The expected balance together with the value of a pre-approved credit line the ASPSP makes permanently available to the user.

For card-accounts:

- "money to spend with the value of a pre-approved credit limit on the card account"

4. **"openingBooked"**: Book balance of the account at the beginning of the account reporting period. It always equals the closing book balance from the previous report

5. **"interimAvailable"**: Available balance calculated in the course of the account servicer's business day, at the time specified, and subject to further changes during the business day. The interim balance is calculated on the basis of booked credit and debit items during the calculation time/period specified

For card-accounts, this is composed of

- invoiced, but not yet paid entries,
 - not yet invoiced but already booked entries
6. "**forwardAvailable

7. "**nonInvoicedOnly for card accounts, to be checked yet****

8. Get Balances

(core/rest/api/accounts/{bankName}/{resourceId}/{internalConsentId}/balances)

Balance can be checked for each account - "resourceId" from GetAccounts must be sent. Parameter "PSU-IP-Address" is mandatory on API headers.

Example of GetBalances headers

The screenshot shows a Postman request configuration. The method is set to 'GET'. The URL is `((API_BASE_URI))/gateway/core/rest/api/accounts/BT/K13RONCRT0060214301/{internalConsentId}/balances`. In the 'Headers' tab, there is one entry: 'PSU-IP-Address' with the value '192.12.123.32'. Other tabs like 'Params', 'Authorization', 'Body', etc., are visible but not active.

GetBalances response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO98BTRLRONCRT0ABCDEFGHI",  
            "bban": "BARC12345612345678", (only for RZB)  
            "currency": null (is null only for BT, FB, REV, OTP, LIB)  
        },  
        "balances": [  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 485.55  
                },  
                "balanceType": "expected",  
                "referenceDate": "2023-02-03",  
                "creditLimitIncluded": "false",  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": null, (This data element might be used to indicate e.g. with the expected or booked balance that no action is known on the account, which is not yet booked)  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            }  
        ]  
    }  
}
```

```

        },
        {
            "balanceAmount": {
                "currency": "RON",
                "amount": 485.55
            },
            "balanceType": "expected",
            "referenceDate": "2023-02-03",
            "creditLimitIncluded": "true",
            "creditDebitIndicator": null,
            "lastChangeDateTime": null,
            "lastCommittedTransaction": null,
            "transactionStatus": null
        }
    ]
}
}

```

9. Get Transactions

(/core/rest/api/accounts/{bankName}/{resourceId}/{internalConsentId}/transactions)

The transactions for each account can be obtained ("resourceId" must be completed from GetAccounts).

Currently, the transactions can be obtained for the last 90 days.

Parameter "PSU-IP-Address" is mandatory on API headers.

The following parameters are requested:

Parameter	BT	BCR	ING	BRD	RZB	CEC	FB	UNC	REV	OTP	LIB	GRT
bookingStatus	M	M	N.R.	M	M	M	M	M	M	M	N.R.	M
dateFrom	M	M	M	M	M	M	M	M	M	M	M	M
dateTo	M	M	M	M	M	M	M	M	M	M	M	M

* M – mandatory

* N.R. – not requested

For Banca Transilvania, the "bookingDate" and "valueDate" have different values for PF and for PJ accounts. For PJ accounts, the "bookingDate" and "valueDate" are completed also with the payment initiation date. But when searching the transactions, if the payment initiation date is different from processing date, then it must be searched after processing date. The same behaviour is in the internet banking application.

Banks which have pagination will send on API response the parameter "next". When "next" has a value, then this means that another page with transactions is available. Furthermore, when "next" is null, then no other page with transactions is available.

If you want to move to the next page with transactions, then you should send on header the "next" parameter and its value (link).

Please note that "next" parameter is correlated with "dateFrom" and "dateTo" and any API call will raise an error if that period is changed. This means that if you want to retrieve transactions from other period, then you should make a new GetTransactions call with the new values for "dateFrom" and "dateTo" so that "next" could work with the new period.

Transactions are sent either bulk or with pagination, as following:

Bank	How the transactions are sent
BT	100 transactions / page
BCR	Bulk
CEC	Bulk
BRD	200 transactions / page
ING	20 transactions / page
REV	50 transactions / page
UNC	400 transactions / page
RZB	50 transactions / page
OTP	Bulk
LIB	Bulk
FB	Bulk
GRT	Bulk

GetTransactions response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO98BTRLRONCRT0ABCDEFGHI",
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "fbced129e8384109ae2023020522000000012", (null
            only for BRD and UNC)
          "entryReference": null,
          "endToEndId": null, (Unique transaction reference given by the initiator of the
            transaction)
          "bookingDate": "2023-02-05", (processing date. AS-IS: all banks except BT will
            sort transactions after bookingDate.)
          "valueDate": "2023-02-05", (initiation date. AS-IS: only BT sort transactions after
            valueDate)
          "executionDateTime": null, (only for ING)
          "transactionAmount": {
            "currency": "RON",
            "amount": 1200.19
          },
        ],
      }
    }
  }
}
```

```

    "transactionType": null,
    "exchangeRate": [], (object can be null depending on transaction's type)
    "creditorName": "Dan Popescu", (it can be null depending on transaction's type)
    "creditorAccount": {
        "iban": "RO98BTRILRONCRT0ABCDEFGHI"
    }, (it can be null depending on transaction's type)
    "debtorName": "Dana Popescu", (it can be null depending on transaction's type)
    "debtorAccount": {
        "iban": "RO98BTRLEURCRT0ABCDEFGHI"
    }, (it can be null depending on transaction's type)
    "remittanceInformationUnstructured": "Merchant reference 202302
0522000000012", (MAX 140 TEXT – different transaction's details from core banking. It will be different from bank
to bank and depends on transaction's type. REV does not send it)
    "proprietaryBankTransactionCode": null, (code or description attributed by
the bank regarding transaction's type)
    "additionalInformation": null, (only for BCR and REV. For BCR it shows an
integer and for REV it shows information similar with the one from parameter
"remittanceInformationUnstructured")
    "transactionStatus": null, (only for REV)
    "creditDebitIndicator": null (only for REV)
},
{transaction no. 2, 3, etc.}
],
"pending": [],
"links": null
},
"balances": []
}
}

```

10. GetTransactionDetails

(/core/rest/api/accounts/{bankName}/{resourceld}/{internalConsentId}/transactions/{transactionId}/details)

This API returns the same information as GetTransactions but only for a specific transaction (you must complete the "transactionId" and "resourceld"). Parameter "PSU-IP-Address" is mandatory on API headers.

Only BT, BCR, CEC and GRT have implemented this API.

Example of GetTransactionDetails headers

The screenshot shows the Postman interface with a GET request. The URL is: `((API_BASE_URI))/gateway/core/rest/api/accounts/BT/K13R0NCRT0060214301/({internalConsentId})/transactions/fbcfed119e8384109ae201911112200000001/details`. The 'resourceld' part of the URL is underlined in red. The Headers tab is selected, showing a single header: 'PSU-IP-Address' with the value '192.168.1.1'. Other tabs like Params, Authorization, Body, Pre-request Script, Tests, and Settings are visible at the top.

* Red underline – "resourceld"

* Green underline – "transactionId"

GetTransactionDetails response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "transactionId": "fbced119e8384109ae2019111122000000011",  
        "entryReference": null,  
        "endToEndId": null,  
        "bookingDate": "2019-07-28",  
        "valueDate": "2019-07-28",  
        "transactionAmount": {  
            "currency": "RON",  
            "amount": 1100.2  
        },  
        "transactionType": null,  
        "creditorName": "Dan Popescu",  
        "creditorAccount": {  
            "iban": "RO98BTRLRONCRT0ABCDEFGH1"  
        },  
        "debtorName": "Dana Popescu",  
        "debtorAccount": {  
            "iban": "RO98BTRLEURCRT0ABCDEFGH1"  
        },  
        "remittanceInformationUnstructured": "Merchant reference 201907282100000001",  
        "executionDateTime": null  
    }  
}
```

11. GetAccountDetails

(/core/rest/api/accounts/{bankName}/{resourceId}/{internalConsentId}/details)

This API is used to obtain account details for a specific account, not for all accounts. ING doesn't provide this API. You must complete the "transactionId". Parameter "PSU-IP-Address" is mandatory on API headers.

Only ING did not implement this API.

Example of GetAccountDetails headers



KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
PSU-IP-Address	192.168.123.32	Description		
Key	Value	Description		

GetAccountDetails response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "resourceId": "K13RONCRT0060214301",
    "iban": "RO98BTRLRONCRT0ABCDEFGHI",
    "currency": "RON",
    "name": "Cont de disponibil",
    "product": "Cont de disponibil",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 485.55
        },
        "balanceType": "expected",
        "referenceDate": "2023-02-06",
        "creditLimitIncluded": "false",
        "creditDebitIndicator": null
      },
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 485.55
        },
        "balanceType": "expected",
        "referenceDate": "2023-02-06",
        "creditLimitIncluded": "true",
        "creditDebitIndicator": null
      }
    ],
    "sortCodeAccountNumber": null
  }
}
```

12. DeleteConsent

This API can be used to revoke access given through a consent. Parameter "PSU-IP-Address" is mandatory on API headers.

Example of DeleteConsent headers

The screenshot shows the Postman interface with a DELETE request to `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/BT/{internalConsentId}`. The Headers tab is active, displaying the following configuration:

KEY	VALUE	DESCRIPTION
PSU-IP-Address	192.12.123.32	Description
Key	Value	Description

DeleteConsent response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": {  
    "message": "Consent has been successfully revoked"  
  }  
}
```

After calling the API DeleteConsent, any other called API will response with the following structure:

```
{  
  "status": 401,  
  "messageStatus": "Unauthorized",  
  "moreDetails": {  
    "text": "TerminatedByTpp"  
  }  
}
```

6. Banks flow

All flows were made using testing accounts on Sandbox environment and data provided by the banks.

For "initConsent" flow (after which PSU will interact with the bank to authorise the consent), if there are UI differences between SANDBOX and PROD environments, we will provide examples from both types of interaction with the bank.

1. Banca Transilvania - Flow

1. Authenticate

The screenshot shows the Postman interface for a 'Authenticate' request. The method is 'POST' and the URL is `((MTLS_BASE_URL))/gateway/authenticate/rest/api/token`. The 'Body' tab is selected, showing a key-value pair: 'client_id' with value `((client_id-partner))`. The 'Tests' tab contains a pre-request script:

```
var jsonData = JSON.parse(responseBody);
postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);
```

The screenshot shows the Postman interface after sending the 'Authenticate' request. The response body is displayed in the 'Tests' tab's pre-request script area:

```
var jsonData = JSON.parse(responseBody);
postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);
```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101660,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTIyMjcsImIhdCI6MTY1NTI5MTkyNywianRpIjoiNWM5MWZlNDUtMjk0Ny00Nzg2LWI5MTYtYjI1ZDZjYjM3NmJlIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOiI2NmE2ZTIyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiN2UyYmU5NDYtZWMzNC00MWJ1LtlhMDQtNTYwMmU2MzQyNDDiIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFSbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VyY2VfYWNjZXNzIjp7InRlc3RjbGllbnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbi
```

```

JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hz2UTYWNg3VudCIsIm1hbmcFnZS1hY2NvdW50LWxpbtzI
iwidm1ldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAx
NjYwIiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjYwIiwiY2xpZW50SWQiOij0ZXN
0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn
ByZWZlcnJ1ZF91c2VybmFtZSI6InNlcnpY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZ
XNzIjoiMTAuMC4yNDguMzIifQ.K901Ya-ceJYwJt0m-
nqIGUk7KG0L589iB7KaH0vbpOQPez_L0snrix7RDZvx09WibXXi2rrM9w6XweBDjxpLXYYEkGIJWfjf-g-
Cy5WDu3JFP_S-
P1mk2dmHyDs5GUvJw3H0dER9uSdkgjqQjPK8JMnI18080vPnZmUAHEI3MFYVPZ583ust45dW5-
1ceLDpPY8F-
nEqWqdoYkj_A_AoR35e4uYfZKBZKJP3uuuPJgwLPvY1UdYOMYEyiw0Fi28K07YVeilGdqsVKnnmW-
11jRje4qLTYswSRSR-Pae8062svnbkzqEjRaz_fYRmCo6JhhsCwNRpSGoxUxf8e_SkoA",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldeIiA6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05Ntc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNjc5MjcsImhdCI6MTY1N
TI5MTkyNywianRpIjoiYTE1NzM4MTctN2IwMy00MmRkLThmYjMtOTBhZWJjNTgzOTNIiwiiaXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6IjdlMmJ1OTQ2LWVjMzQtNDFiZS05YTA0LTU2
MDJ1NjM0MjQ3YiIsInNjb3B1IjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY2MCJ9.f4
o8vBO9dkMGnw2RX0MRsQR8Te1Z_vUeWowfkjRuE0Y"
}
}

```

2. initConsent

KEY	VALUE	DESCRIPTION	... Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json			
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
<input type="checkbox"/> Key	Value	Description		

```

1
2
3
4
5
6
7
8
{
  "codeBank": "BT",
  "periodOfValid": 90,
  "psuEmail": "catalin.zaman@itsmartsystems.eu",
  "psuIntermediaryId": "123",
  "redirectURL": "https://www.google.com",
  "TCaccepted": true
}

```

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "consentStatus": "RECEIVED",  
        "consentId": null,  
        "_links": {  
            "scaOAuth": {  
                "href": "https://apistorebt.ro/mga/sps/oauth/oauth20/authorize?stat  
e=eyJlbmMioiJBMTI4Q0JDLUhTMju2IiwiYWxnIjoiZGlyIiwiY3R5IjoiSldUIn0..8NZgOhkz1NK3-  
XxGPFDhfA.AaoyUSHt0QRA8od2wnF0XGSJ2V8jqctcU0G8o16GI70JKaTk63dww6lc4fEKhhYpW1QZgwcAk  
b65DVVFjun96N78acNBaVmwJSL0ACC42-  
MeNFSul3dxHebiSEhTIQqSTjjcs9YzKhYRNAYwhEmpCE8iaDQoeZDePjrFdK4CXcSKFvI_ai_fah91wJwe  
0ie6RNvw0o7ziLre1JKFXTYLvGzEmtG1khJfDMg1AxZiUbarJC0PAjPbT6LCDbvQSIBtO08I9Trw65qsW18  
FpJ6Zhzi0Ivn5SmUsAN4te6jLGDRYa9-UejYdo-  
sZoO14yyErmhuTF461veP5R71MLXa6OUhk7Z0bOfS39mC6nxycsweVP4EAOTcGis3p7UNz61QjyVAxwFU1  
UZq2TSrAbVRg1Y3EtsCHyJJYjqB4rcsboWmE8rPuhNJ6fRhbec2YuUYCJeZgnRCNPkrUfCDElNEpBqAvg4H  
HMcLgwxMU1aG0byf5poiXPRTTuT35N3wXPJoGVSlmmFJ7OBOC8MsgvYt_8pwnWqCO5Os81eJuuqYi7-  
MrAjdpVA4XFzqCyMBR9tM1ZE8Q1rVqwdke70__DzrSVED4IE3Idqyopw2VfEc5sjaW-  
RpyKeqpql8R1cJMJekPzbXJi0NwHyLZZOJmjFdtGhYKBgrOvUF17VJy6H0Dgt56yi8FITXY6BOim95Ey1Xf  
Zz_JHzC3UH8Kyssw3I46PLlsruG7YcfPzMzs4G7hoYCx9P3Hgb1f8SCs3U0sAAAsfMJ1aVLq6EMuS0hiTXR6a  
90sE0a5gsejbyQDWY7VUFB6-GudcTafUV1xL8faE01gM-WzvfPrPZbLO-  
Vosjh2WV5n2YBtg17N7rM5eI93SK1IHYWLVLxKroes-2ZTrLxNJGf4-s_9Fx-  
SryW3eu6zi5s746epMt7sTStzlihvdf2eG4qKBWdp5JyPpb4cm3xAutRWcMzVbW6HqrRsSWZLroX9ZchQPf  
g6fafawpoMTf34qIdBHxBWF3AzYjKjH4QkWOQJ5qH-  
iWLRTSqdn7AMoQyvn4xHh7lpE_HYAkKqzLB3FTHeRSoPoMR6m-  
92EHffFCOqkCF5TNH_rcKrawzfaMsTw.FFV_ZGJQ2AiEgNVd4V6_jQ&response_type=code&client_id=  
a36732b0-b710-4a14-813a-  
88d0e3f3383a&redirect_uri=https://api.sandboxaccounts.smartfintech.eu/gateway/core/  
rest/api/confirmation&scope=AIS:aic5d3c1e29ed16714bbf488red&code_challenge=J1rlc1eI  
UkGzSU6qbUTL0-  
JMUGLf_XQ1N3Jn8DmcI3w&code_challenge_method=S256&nonce=nonce_1655297699218"  
    }  
}  
}
```



Authentication

Choose the login method



[I forgot my NeoBT password](#)



Authentication

Choose the login method



For sure, check the first 2 digits again. They're the same? If they are not identical, do not add the code and call us at 0264 308 028.

Add the code received via SMS:

-

[BACK](#)

0264 308028 • contact@btrl.ro



Through the Third Party Payment Services Provider platform **SmartAccounts DEMO** you are going to view the IBANs, account balance and transaction history for 90 days, from your following payment accounts opened with BT:

Select accounts

	Current account	RO98BTRLRONCRT0ABCDEFGH
	Current account	RO98BTRLEURCRT0ABCDEFGH

Select all accounts

If you are an individual customer, please note that your personal data regarding the accounts' IBANs, balance and transaction history will be processed by BT and will be disclosed to the Third Party Payment Services Provider.

RETURN TO TPP

After selecting accounts and pressing "RETURN TO TPP", thus authorising the consent for selected account(s), PSU or developer will be redirected to the established page.

3. GetConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 1. Intermediary API's BT / 5. GetConsent

GET {{{API_BASE_URI}}}/gateway/core/rest/api/consent/BT/{{internalConsentId}} Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

Headers 7 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.168.1.123		...		
<input checked="" type="checkbox"/> Content-Type	application/json		...		
Key	Value	Description	...		

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO98BTRLRONCRT0ABCDEFGH"
        }
      ]
    }
  }
}
```

```

        },
        {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        }
    ],
    "balances": [
        {
            "iban": "RO98BTRLRONCRT0ABCDEFGH"
        },
        {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        }
    ],
    "transactions": [
        {
            "iban": "RO98BTRLRONCRT0ABCDEFGH"
        },
        {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        }
    ],
    "availableAccounts": null
},
"consentStatus": "valid",
"frequencyPerDay": 4,
"lastActionDate": "2023-02-07",
"validUntil": "2023-03-09",
"recurringIndicator": true,
"combinedServiceIndicator": null,
"creationDate": null
}
}
}

```

4. GetConsentStatus

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3' with a specific request labeled '6. GetConsentStatus'. The request method is 'GET' and the URL is `{{API_BASE_URI}}/gateway/core/rest/api/consent/BT/({internalConsentId})/status`. The 'Headers' tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json			
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {

```

```

        "consentStatus": "valid"
    }
}

```

5. GetAccounts

The screenshot shows the Postman interface with the following details:

- Request URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/BT/{internalConsentId}
- Method:** GET
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Params:** None
- Authorization:** Bearer token (not shown)
- Body:** None
- Pre-request Script:** None
- Tests:** None
- Settings:** None

Response:

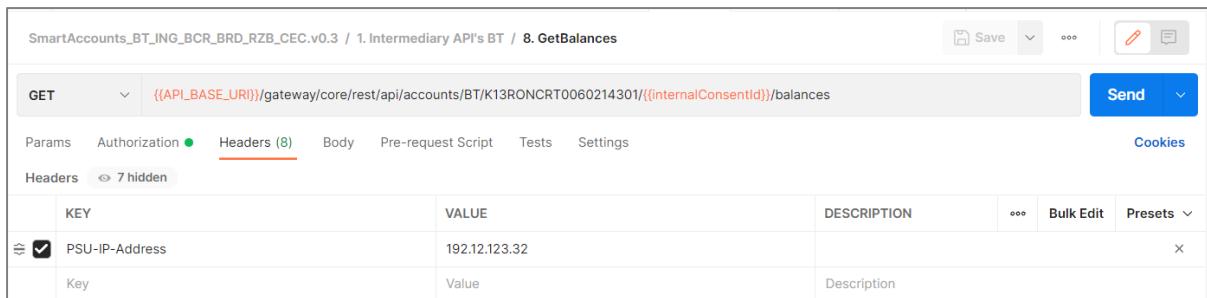
```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "K13RONCRT0060214301",
      "iban": "RO98BTRLRONCRT0ABCDEFGHI",
      "currency": "RON",
      "name": "Cont de disponibil",
      "product": "Cont de disponibil",
      "cashAccountType": "CurrentAccount",
      "status": null,
      "bic": null,
      "accountOfficerDescription": null,
      "openingDate": null,
      "customerSector": null,
      "customerType": null,
      "accountUsage": null,
      "workingBalance": null,
      "onlineActualBalance": null,
      "balances": [
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 485.55
          },
          "balanceType": "expected",
          "referenceDate": null,
          "creditLimitIncluded": "false",
          "creditDebitIndicator": null
        },
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 485.55
          }
        }
      ]
    }
  ]
}
```

```

        },
        "balanceType": "expected",
        "referenceDate": null,
        "creditLimitIncluded": "true",
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
},
{
"resourceId": "K13EURCRT0060214301",
"iban": "RO98BTRLEURCRT0ABCDEFGH",
"currency": "EUR",
"name": "Cont de disponibil",
"product": "Cont de disponibil",
"cashAccountType": "CurrentAccount",
"status": null,
"bic": null,
"accountOfficerDescription": null,
"openingDate": null,
"customerSector": null,
"customerType": null,
"accountUsage": null,
"workingBalance": null,
"onlineActualBalance": null,
"balances": [
{
    "balanceAmount": {
        "currency": "EUR",
        "amount": 1156.51
    },
    "balanceType": "expected",
    "referenceDate": null,
    "creditLimitIncluded": "false",
    "creditDebitIndicator": null
},
{
    "balanceAmount": {
        "currency": "EUR",
        "amount": 1156.51
    },
    "balanceType": "expected",
    "referenceDate": null,
    "creditLimitIncluded": "true",
    "creditDebitIndicator": null
}
]
},
"sortCodeAccountNumber": null
}
]
}

```

6. GetBalances



The screenshot shows the Postman interface with a GET request to `/accounts/BT/K13RONCRT0060214301/({internalConsentId})/balances`. The Headers tab is selected, displaying the following configuration:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32		X		
Key	Value	Description			

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO98BTRLRONCRT0ABCDEFGHI"  
        },  
        "balances": [  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 485.55  
                },  
                "balanceType": "expected",  
                "referenceDate": "2023-02-07",  
                "creditLimitIncluded": "false",  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": null,  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            },  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 485.55  
                },  
                "balanceType": "expected",  
                "referenceDate": "2023-02-07",  
                "creditLimitIncluded": "true",  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": null,  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            }  
        ]  
    }  
}
```

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 1. Intermediary API's BT / 9. GetTransactions

Save ⚙️ ... Send

GET {{API_BASE_URL}}/gateway/core/rest/api/accounts/BT/K13RONCRT0060214301/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-01-01&dateTo=2023-01-02

Params ● Authorization ● Headers (9) Body Pre-request Script Tests ● Settings Cookies

```
1 const responseJson = pm.response.json();
2 let next = responseJson.payload.transactions.links;
3 if (next != null) {
4 | postman.setEnvironmentVariable("next_page", next.next.href);
5 }
6 else {
7 | postman.setEnvironmentVariable("next_page", null);
8 }
```

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO98BTRLRONCRT0ABCDEFGH"  
        },  
        "transactions": {  
            "booked": [  
                {  
                    "transactionId": "fbced129e8384109ae202302062200000012",  
                    "entryReference": null,  
                    "endToEndId": null,  
                    "amount": {  
                        "value": 10000,  
                        "currency": "RON"  
                    },  
                    "counterparty": {  
                        "name": "John Doe",  
                        "iban": "RO98BTRLRONCRT0ABCDEFHI",  
                        "bic": "BTRLROBU",  
                        "bicLabel": "BTRL ROMÂNIA BANCĂ",  
                        "branch": "București",  
                        "branchLabel": "BUCURESTI",  
                        "branchCode": "BTRL",  
                        "branchAddress": "Strada Bulevardul Nicolae Ceaușescu 100",  
                        "branchCity": "București",  
                        "branchCountry": "Romania",  
                        "branchPostCode": "010000",  
                        "branchPhone": "+40 21 1234567",  
                        "branchEmail": "john.doe@bank.ro",  
                        "branchFax": "+40 21 1234567",  
                        "branchUrl": "http://www.bank.ro/john-doe"  
                    }  
                }  
            ]  
        }  
    }  
}
```

```

        "bookingDate": "2023-02-06",
        "valueDate": "2023-02-06",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "RON",
            "amount": 1200.19
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": "Dan Popescu",
        "creditorAccount": {
            "iban": "RO98BTRLRONCRT0ABCDEFGH"
        },
        "debtorName": "Dana Popescu",
        "debtorAccount": {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        },
        "remittanceInformationUnstructured": "Merchant reference 202302
0622000000012",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    },
    {
        "transactionId": "fbced139e8384109ae2023020622000000013",
        "entryReference": null,
        "endToEndId": null,
        "bookingDate": "2023-02-06",
        "valueDate": "2023-02-06",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "RON",
            "amount": 1300.18
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": "Dan Popescu",
        "creditorAccount": {
            "iban": "RO98BTRLRONCRT0ABCDEFGH"
        },
        "debtorName": "Dana Popescu",
        "debtorAccount": {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        },
        "remittanceInformationUnstructured": "Merchant reference 202302
0622000000013",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
}

```

```

        }
    ],
    "pending": [],
    "links": {
        "next": {
            "href": "aHR0cHM6Ly9tdGxzLmFwaS5zbWFydHBnLmV1L3NtYXJ0ZmludGVjaC9zYW5kYm94L3BzZDIyWlzcC92MS9hY2NvdW50cy9CVC9LMTNST05DULQwMDYwMjE0MzAxL3RyYW5zYWN0aW9ucz9kYXR1RnJvbT0yMDIzLTAXLTA2JmRhGvUbz0yMDIzLTAYLTA3JmJvb2tpbmdTdGF0dXM9Ym9va2VkJnJ1Y29yZHNQZXJQYWdlPTEwMCZwYWdlTnVtYmVyPTI="
        }
    }
},
"balances": []
}
}

```

8. GetTransactionsDetails

The screenshot shows a REST API client interface with the following details:

- URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/BT/K13RONCRT0060214301/{{internalConsentId}}/transactions/fbcded119e8384109ae2019111122000000011
- Method:** GET
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Buttons:** Save, Send, Cookies

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "transactionId": "fbced119e8384109ae2019111122000000011",
        "entryReference": null,
        "endToEndId": null,
        "bookingDate": "2019-07-28",
        "valueDate": "2019-07-28",
        "transactionAmount": {
            "currency": "RON",
            "amount": 1100.2
        },
        "transactionType": null,
        "creditorName": "Dan Popescu",
        "creditorAccount": {
            "iban": "RO98BTRLRONCRT0ABCDEFGH"
        },
        "debtorName": "Dana Popescu",
        "debtorAccount": {
            "iban": "RO98BTRLEURCRT0ABCDEFGH"
        },
    }
}
```

```

        "remittanceInformationUnstructured": "Merchant reference 201907282100000001
1",
        "executionDateTime": null
    }
}

```

9. GetAccountDetails

The screenshot shows a Postman collection named "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3" with a single request titled "11. GetAccountDetails". The request is a GET to `{{API_BASE_URI}}/gateway/core/rest/api/accounts/BT/K13RONCRT0060214301/({internalConsentId})/details`. The "Headers" tab is active, displaying the following header:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "resourceId": "K13RONCRT0060214301",
    "iban": "RO98BTRLRONCRT0ABCDEFGHI",
    "currency": "RON",
    "name": "Cont de disponibil",
    "product": "Cont de disponibil",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 485.55
        },
        "balanceType": "expected",
        "referenceDate": "2023-02-07",
        "creditLimitIncluded": "false",
        "creditDebitIndicator": null
      },
      {
        "balanceAmount": {

```

```

        "currency": "RON",
        "amount": 485.55
    },
    "balanceType": "expected",
    "referenceDate": "2023-02-07",
    "creditLimitIncluded": "true",
    "creditDebitIndicator": null
}
],
"sortCodeAccountNumber": null
}
}
}

```

10. DeleteConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 1. Intermediary API's BT / 12 DeleteConsent

DELETE https://{{API_BASE_URI}}/gateway/core/rest/api/deleteConsent/BT/{{internalConsentId}}

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers (8 hidden)								
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	PSU-IP-Address		192.12.123.32					
		Key	Value		Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "message": "Consent has been successfully revoked"
  }
}
```

11. RefreshToken

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 1. Intermediary API's BT / 1.1 RefreshToken

POST https://{{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Send

Params	Authorization	Headers (10)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers (8 hidden)								
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	internalConsentId		{{internalConsentId}}					
<input checked="" type="checkbox"/>	access_token		{{access_token}}					
		Key	Value		Description			

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 103812,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJMZTVKbU
JFWlJONWYxSWxQQ1hBQWdVMWpQYmh1b1FwoEJHUj1KRjAtTmZjIn0.eyJleHAiOjE2NzU3ODEyMTUsImhd
CI6MTY3NTc4MDkxNSwianRpIjoiZmRhMzQ3NzgtYjNkOC00MzYwLWFmYmQtMzZmMTYyZWJhNTkwIiwiaXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOijJnjFjODc1MC01MTNkLTQxNGYtOGQ5Ny1mZDY3NjMwYzdkZGUilC
J0eXAiOjJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaWuX3N0YXR1IjoiNDc2MzUwM2ItM
DNiNi00MjdilThkYWEty2Q2MGRlZWYxNzE1IiwiYWNyIjoiMSIiImFsbG93ZWQtb3JpZ2lucyI6WyIqI10s
InJlyWxtX2FjY2VzcyI6eyJyb2xlcI6WyJvZmZsaW51X2FjY2VzcyIsInVtYV9hdXRob3JpemF0aW9uI11
9LCJyZXNvdXJjZV9hY2N1c3MiOnsiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCIiIm1hbm
FnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2Nvb
nNlbnRfaWRfY2xhaW06MTAzODEyIiwigc21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAzODEy
IiwiY2xpZW50SG9zdCI6IjEwLjAuMjExLjc1IiwiY2xpZW50SWQioiJ0ZXN0Y2xpZW50djEiLCJ1bWFpbF9
2ZXJpZmllZCI6ZmFsc2UsInByZWlcnJlZF91c2VybmFtZSI6InNlcnPjY2UtYWNjb3VudC10ZXN0Y2xpZW
50djEiLCJjbG11bnRBZGRyZXNzIjoiMTAuMC4yMTEuNzUifQ.S7YGROFn82ziOdwccCseHdPM3PkCn6tjf-
A-
F33pVatgLmPoA1xnQDnAznaDgG8QfZcxC3CEiis1PW61i9dg2sEDFM7yGeXu8g7dRCuIne28JAeWjh1HIZi
99wyqc4YOHYfb3oj1rpSMB3_-dtAXqFlotzIzejlJKbs-
svCMMGehLVIALdm2vD2kumoAQyow3csZh0elGlYr-QI-
rd66KD6SHXMtlvBS1ST1qjSYYN7afRaOk49aZKP8bi134rBQHLboDph9Q_QEOH1eS3z5VdOE6bW_Ga3R8We
abrCiyC6zO-EKFjO_p0zVvpety8leTfgBcNXygW6Dn_c64Ncwjw",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC1I1NjUwN
jNmYi05ZDJ1LTQ1ZTgtOTY1OC02OWZhZTE0ZTEwODAifQ.eyJleHAiOjE2ODM1NTE3MzUsImhdCI6MTY3N
Tc4MDkxNSwianRpIjoiZTI2ZGYwZDYtNzFmMC00ZTNjLTk4YmYtMjcyOTg3NTVhMTMxIiwiaXNzIjoiHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyIsImFlZCI

```

```

6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOijJnjFjODc1MC01MTNkLTQxNGYtOGQ5Ny1mZDY3NjMwYzdkZGUilCJ0eXAiOiJPZmZsaW5liwiY
XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc2lvb19zdGF0ZSI6IjQ3NjM1MDNiLTAzYjYtNDI3Yi04ZGFhLWNk
NjBkZWVmMTCxNSIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjEwMzgxMiJ9.Ge
1I6-6j_zt1JYFTqrPTMnR4LWv42h4Ops33MMsZMuQ"
}
}

```

In case the "accessToken" expires, the response will have the following structure:

```

{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}

```

2. ING Bank – Flow

1. Authenticate

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> client_id	{{{client_id-partner}}}			
<input type="checkbox"/> client_secret	{{{client_secret-partner}}}			
Key	Value	Description		

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101686,
    ...
  }
}

```

```

    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0eyJleHAiOje2NTUyOTg3MzcsImhd
CI6MTY1NTI5ODQzNywianRpIjoiZTU4ZmFlyTAtNmIyNC000DBmLTk1ZDYtY2JmZWQxMmUwYWY4Iiwi
aXNz
IjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cy
IsImF1ZCI6ImFjY291bnQilCJzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWfMz1mOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWFyZXIiLCJhenAiOj0ZXN0Y2xpZW50djeiLCJzZXNzaW9uX3N0YXR1IjoiNTFiN2E3ZDAtY
jE0YS00YjhkLWE4YzEtMzI1MmFkZjYxMGQwIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKijD
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZv9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG1bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbi
JdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtyW5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidm11dy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydgFjY291bnRzX2NvbnN1bnRfaWRfY2xhaW06MTAx
Njg2IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjg2IiwiY2xpZW50SWQioiJ0ZXN
0Y2xpZW50djeiLCJjbG1bnR1b3N0IjoiMTAuMC4yNDguMzIifQ.cCvndmaj7wK0ARsf5qeliL4hz8iJHlgcJcD8QG-
X1rqVc49RjibavdQL89uAr6026jB1OY4bavN1qhlbuVjfTE9fIaz5XKTaz2I8QpprlrxQ1wG95dZDD4GKXPz
3plaRCFOWhXVeZnXg5rEZqn9iQvOatOpXctZe8D8XDoxh-
ZmN9TCdE2oCGujk25uOmrdc6GdfSoutzd2L0ry4LqtAbsyXfFNJL6SiVFE1htHk47idnTxixX-
WjlnyuMbVtMNuIKmV6ORbTR8SL26VOcS5oey2I1B6wBLfMCxyPMRlxZ-_fYxmrcogu9T-
rdHXOIxoDyFqFHMIo8xRjNNCIgAIg",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl3OWNhY
2RmMi1kMjk3LTQ0YWIYjY5Ni05NtC00DRmYTU2ZDyifQ.eyJleHAiOjE2NjMwNzQ0MzcsImhdCI6MTY1N
TI5ODQzNywianRpIjoiZjczNjZkNjAtMmIzOC00NjhjLTkzzjMtMzU4Mjg1ODg4MWIxIiwi
aXNzIjoi
aHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWfMz1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpWIjoidGVzdGNsaWvudHYxIiwiC2Vzc21vb19zdGF0ZSI6IjUxYjdhN2QwLWIxNGEtNGI4ZC1hOGMxLTMy
NTJhZGY2MTBkMCIsInNjb3BlIjoi
c21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY4NiJ9.Gp
713_BnvYKaegI4H4R0741Mg7IZdJXBjZGKaOFFW6A"
}
}

```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 4. InitConsent

POST https://{{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}} Send

Params	Authorization	Headers (12)	Body	Pre-request Script	Tests	Settings	Cookies
		Headers					
Headers	9 hidden						
KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets		
<input checked="" type="checkbox"/> Content-Type	application/json						
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32						
<input type="checkbox"/>							
Key	Value	Description					

The screenshot shows a Postman interface with the following details:

- Request Method:** POST
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}
- Body:** JSON (selected)


```

1
2   "codeBank": "ING",
3   "periodOfValidity": 90,
4   "psuEmail": "catalin.zaman@itsmartsystems.eu",
5   "psuIntermediaId": "123",
6   "redirectURL": "https://www.google.ro/",
7   "TCaccepted": true
8
      
```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://api.sandboxaccounts.smartfintech.eu/gateway/core/rest/api/authorization?internalConsentId=101686&bankRedirectUri=https://myaccount.sandbox.ing.com/granting/2e79dfe2-64f4-4274-abfb-fa5b06f804e4?state%3DZXlKcmFXUw1PaU14TVdFNVl6VmllaTB6TmlVeUxUUahaRF10T1RVek1DMDJORFJsTldFeE56ZGtNelFpTENKaGJHY2lPaUpTVXpJMUs5pSjkuZXlKbGVIQwlPakUyTlRVek1ESXdORGtzSW1saGRDSTZNVFkxTlRJNU9EUTBPU3dpY3lJNkldrRkpVeUlzSW1NaU9pSXlaVGM1WkdabE1pMDJOR1kwTFRReU56UXRZV0ptWWkxbV1UVmlNRFptT0RBMFpUUWlmUS50R3dFTTdXRDJNWU1ELUhBdGdJWjVYNEctX3p0WXA3Y2FmQzhDX01TLUowOFCzXzNpUjNVd2pyeWtaVEFOMhLdjhoa3lnVUNOb1F5d1NhT1BCV0Y1LUhidGFMMHBRdnhWcC1zWGhWZ05QdGZXEo4NVl5Q1RNZnowQmx3NTR6eHBKSjVZMHBNT2VrOHUxRGVYeWVQMWFibT1HNkZwdGxLWkVFekI5LXJGVFpGSV1rcjZaT0VMM0xCeHJCelVEY2pYWldTWmRfNURyejdyc3BrVFZYOTzwWWIzMTR4c0FFTk1RVjhXaXVxS2xRcmQ1bW9Mzm9Id2ZmTkNVZW1UTWY0SjBXdmtIVjRNNkVxZDY5NFRyTndJajFOR1cxVXBMcHFOYnpjT0NSZDNsLxo5U1JPUVRqWDlqZG1oYWR2cGNaTk5YMkVHN1J3Z2UxYW51a2RPNUlhQ2c%3D%26client_id%3D5calable-c0ca-c01a-cafe-154deadbea75%26redirect_uri%3Dhttps://api.sandboxaccounts.smartfintech.eu/gateway/core/rest/api/confirmation%26scope%3Dpayment-accounts:balances:view%20payment-accounts:transactions:view%26response_type%3Dcode"
      }
    }
  }
}
```

2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application

Welcome to your bank

Country of your ING account

ING Romania

Cancel

Next

With this permission you allow

- View balance information for a payment account
- View transaction information for a payment account

Test profiles

Please select test profile

Popescu Corina
payment-accounts:balances:view
payment-accounts:transactions:view

A Van Dijk
payment-accounts:balances:view
payment-accounts:transactions:view

Alice and Bob
payment-accounts:balances:view
payment-accounts:transactions:view

Jane Doe
payment-accounts:balances:view
payment-accounts:transactions:view

John Doe
payment-accounts:balances:view
payment-accounts:transactions:view

Peter Smith
payment-accounts:balances:view
payment-accounts:transactions:view

Sarah Smith
payment-accounts:balances:view
payment-accounts:transactions:view

W Van Dijk
payment-accounts:balances:view
payment-accounts:transactions:view

Cancel

Next

After selecting accounts and pressing "NEXT", thus authorising the consent for selected account, developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application

i Verifică întotdeauna că adresa de internet începe cu
<https://myaccount.ing.com>

i Prima dată când folosești contul tău Home'Bank altundeva? Stai fără griji, vei fi asistat pas cu pas pentru configurarea contului.

Bine ai venit! Intră în contul tău ING

ING Home'Bank
 ING Business

Nume utilizator Home'Bank

Parola Home'Bank

Login  Ai nevoie de ajutor? Sună la [031 406 24 64](tel:0314062464)

Acordă acces

i Permisivarea de a accesa anumite conturi de plată va fi valabilă pentru **90 zile**.



Ești pe cale să permiti **SMART FINTECH S.R.L.** accesul la:

- Tranzacții anterioare
- Soldul contului tău

Selectează cont(uri)

Co RO

Co RO

Co RO

Renunță **Continuă**



After inserting the code received via SMS and pressing "Aprobă", thus authorising the consent for selected account(s), PSU will be redirected to the established page.

3. GetConsent

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32				
Content-Type	application/json				x
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [],
      "balances": [],
      "transactions": [],
      "availableAccounts": "allAccounts"
    },
    "consentStatus": "Valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-07",
    "validUntil": "2023-05-07",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

```

        }
    }
}
```

4. GetConsentStatus

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 6. GetConsentStatus

GET {{API_BASE_URI}}/gateway/core/rest/api/consent/ING/{{internalConsentId}}/status

Send

Params	Authorization	Headers (9)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers	7 hidden							
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	Content-Type		application/json					
<input checked="" type="checkbox"/>	PSU-IP-Address		192.12.123.32					
	Key	Value	Description					

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "Valid"
  }
}
```

5. GetAccounts

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 7. GetAccounts

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/ING/{{internalConsentId}}

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers	7 hidden							
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	PSU-IP-Address		192.12.123.32					
	Key	Value	Description					

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "37608c1d-4d6a-40a2-850c-2462f2e26e80",
      "iban": "RO44INGB0000999908767270",
      "currency": "RON",
      "name": "Popescu Corina",
      "product": "Compte Courante",
      "cashAccountType": null,
      "status": null,
      "bic": null
    }
  ]
}
```

```

        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    },
    {
        "resourceId": "1e19978f-b07c-4018-960f-0150bcefd317",
        "iban": "",
        "currency": "EUR",
        "name": "Popescu Corina",
        "product": "Compte Courante",
        "cashAccountType": null,
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 8. GetBalances

GET [\({{API_BASE_URI}}\)/gateway/core/rest/api/accounts/ING/1e19978f-b07c-4018-960f-0150bcefd317/{{internalConsentId}}/balances](https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/ING/1e19978f-b07c-4018-960f-0150bcefd317/{{internalConsentId}}/balances)

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies	
		Headers	7 hidden					
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	PSU-IP-Address	192.12.123.32						
	Key	Value			Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": ""
        }
    }
}
```

```

        "currency": "EUR"
    },
    "balances": [
        {
            "balanceAmount": {
                "currency": "EUR",
                "amount": 1073.86
            },
            "balanceType": "expected",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null,
            "lastChangeDateTime": "2019-02-22 09:16:54",
            "lastCommittedTransaction": null,
            "transactionStatus": null
        }
    ]
}
}

```

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 9. GetTransactions

GET https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/ING/37608c1d-4d6a-40a2-850c-2462f2e26e80/{{internalConsentId}}/transactions?dateFrom=2...

Headers (9)

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32			
next	{{next_page}}			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 2. Intermediary API's ING / 9. GetTransactions

GET https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/ING/37608c1d-4d6a-40a2-850c-2462f2e26e80/{{internalConsentId}}/transactions?dateFrom=2...

Headers (9)

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
dateFrom	2022-05-10		
dateTo	2022-05-12		
Key	Value	Description	

```

const responseJson = pm.response.json();
let next = responseJson.payload.transactions.links;
if (next != null) {
    postman.setEnvironmentVariable("next_page", next.next.href);
}
else {
    postman.setEnvironmentVariable("next_page", null);
}

```

Response:

```

{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO44INGB0000999908767270",
      "currency": "RON"
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "1225",
          "endToEndId": "NOTPROVIDED",
          "bookingDate": "2021-03-30",
          "valueDate": null,
          "transactionAmount": {
            "currency": "RON",
            "amount": -100.0
          },
          "transactionType": "Cash withdrawal",
          "exchangeRate": [],
          "creditorName": null,
          "creditorAccount": {
            "iban": null
          },
          "debtorName": "",
          "debtorAccount": {
            "iban": ""
          },
          "remittanceInformationUnstructured": "Unstructured remittance information example",
          "proprietaryBankTransactionCode": null,
          "additionalInformation": null
        }
      ],
      "pending": [],
      "links": null
    }
  }
}

```

}

8. DeleteConsent

The screenshot shows a Postman request configuration for a DELETE operation. The URL is `{{{API_BASE_URI}}}/gateway/core/rest/api/deleteConsent/ING/{{internalConsentId}}`. The Headers tab is selected, showing two entries: `PSU-IP-Address` with value `192.12.123.32` and a key entry with value `Value`.

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<code>PSU-IP-Address</code>	<code>192.12.123.32</code>			
Key	Value	Description		

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "message": "Consent has been successfully revoked"
  }
}
```

9. RefreshToken

The screenshot shows a Postman request configuration for a POST operation. The URL is `{{{MTLS_BASE_URI}}}/gateway/authenticate/rest/api/refreshToken`. The Headers tab is selected, showing two entries: `internalConsentId` with value `{{{internalConsentId}}}` and `access_token` with value `{{{access_token}}}`.

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<code>internalConsentId</code>	<code>{{{internalConsentId}}}</code>			
<code>access_token</code>	<code>{{{access_token}}}</code>			
Key	Value	Description		

The screenshot shows a Postman request configuration for a POST operation. The URL is `{{{MTLS_BASE_URI}}}/gateway/authenticate/rest/api/refreshToken`. The Body tab is selected, showing two entries: `refresh_token` with value `{{{refresh_token}}}` and `client_id` with value `{{{client_id-partner}}}`.

KEY	VALUE	DESCRIPTION	Bulk Edit
<code>refresh_token</code>	Text <code>{{{refresh_token}}}</code>		
<code>client_id</code>	Text <code>{{{client_id-partner}}}</code>		
Key	Value	Description	

```

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken
Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies
1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101686,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZZJIn0.eyJleHAiOje2NTUyOTk3MTksImIhd
CI6MTY1NTI5OTQxOSwianRpIjoiOTcyZjg0OWQtMjczNS00ZjMxLWE4OWUtZWY1OGQ2ZGM2YzdIIiwiaXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTiyMC11LOWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfYzxIiLCJhenAiOj0ZN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNTFiN2E3ZDAtY
jE0YS00YjhkLWE4YzEtMzI1MmFkZjYxMGQwIiwiYWNyIjoiMSIiImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbi
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UTYWNjb3VudCisImIhbmcFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1Ii19fSwic2NvcGUiOjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAx
Njg2Iiwig21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjg2IiwiY2xpZW50SWQiOj0ZN
0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZm11ZCI6ZmFsc2UsIn
ByZWZ1cnJ1ZF91c2VybmtZSI6InNlcnZpY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZ
XNzIjoiMTAuMC4yNDguMzIifQ.ZMVGPuDdJc8_1CXMKLnazhZPejczY02QVk-
FgyTSNa1hmvlDbC9136Imbfw1noF5NTXTFRt-
lSS1CqVy3q8rFjwpwIXF70G4vct8gbCzkwGX9GS1co8aNczusYfDEaL11ggpYqqo0Tk1LKbSdCNmhL7F8P
LZ9_7ZqJqYR4XC3gt0Q_6DEQtzki7E2a5Eti63gbp161eAnQ3z1Ux6x1eQ-
yHm1kKy0J9bagstcf8X6ICDyGGNmcHjr01j49qlCYoPcfT4YZC2pQqmWlwdakNcnhfLHVERPaxpfcrTk5Sh
aRP_J_-cCJEQ7Fy0N1R7Sk3AcB7fpt8a9eqsPlqkby_Q",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOje2NjMwNzQ0MzcsImIhdCI6MTY1N
TI5OTQyMCwianRpIjoiYTQxZjR1YzMtODI0YS00Y210LWE5YmEtMjIwMTnjZWVmNGUXIiwiaXNzIjoiiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZX1jbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTiyMC11LOWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwig2Vzc21vb19zdGF0ZSI6IjUxYjdhN2QwLWIxNGEtNGI4ZC1hOGMxLTMy
NTJhZGY2MTBkMCIsInNjb3BlIjoiic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY4NiJ9.4Q
EDD51K5qnkU1mo2_vcAeobI9GV3o6luiWJIDrnEpU"
  }
}

```

In case the "accessToken" expires, the response will have the following structure:

```

{
  "status": 401,

```

```

    "messageStatus": "Unauthorized",
    "moreDetails": {
        "text": "Invalid access token"
    }
}

```

3. Banca Comerciala Romana - Flow

1. Authenticate

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 1. Authenticate

POST `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/token`

Body (form-data)

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> client_id	<code>{{client_id-partner}}</code>		
<input type="checkbox"/> client_secret	<code>{{client_secret_partner}}</code>		
Key	Value	Description	

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 1. Authenticate

POST `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/token`

Tests

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 101688,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSlldUIiwia2lkIiA6IClI4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXFiMi1NUUF4SEJXYmlxb2ZJIn0eyJleHAIoJe2NTUyOTk5MjMsImIhd
CI6MTY1NTI5OTYtMywianRpIjoiNWU0NzBkMmEtYzk0OS00NzUxLTlkODEtYTVlM2Y4NTA5ZWfjIiwiAXNz
IjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aNWn1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNTA1MGQ0YmYtM
TQ2MS00ZTg4LWFmNTYtNWUzN2VjZTNiNjc4IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjD
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbij
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG1lbmR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hz2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAx
Njg4IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjg4IiwiY2xpZW50SWQioiJ0ZXN
0Y2xpZW50djEiLCJjbG1lbmRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn
ByZWZlcnJlZF91c2VybmFtZSI6InNlcnPZpY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG1lbmRBZGRyZ

```

```

XNzIjoiMTAuMC4yNDguMzIifQ.LcE3b5eJDFrx3hMvVtqUspQEcq1ZeAC__H9frZ4ZSFTbOCY0L2mym1JA1
t6DzuiroXdmc90JSrgcws1K-
R6NJQ6UlPPi204Znf8vmuls0XOObvvkHBGrjqw4lqyYHIAAtMLUXtoKfJw8AYrKAPQxUOGw3HVZQyHUrttJe
DXS4rLBPh1PtLcF4bvW1W8ato70TQflt3fYYQrXSHJlgG8XBmL7ULqYAXHfdEvhcb--
hQz9eT_yLYjWs54Hb3yrI0f7oT-cumjDmzi-
CwhgGoCFQQucdULp5Q0RAZ3dk9c5fatbRu8wH0cD1Qa52XC2YY6qEO7reOPqNt09x_9331Gs1cQ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzU2MjMsImhdCI6MTY1N
TI5OTYyMywianRpIjoiMDFkYTk5NWItYT11ZS00YTJkLTljNWYtYTdkNzA5Njk0MWE0IiiniaXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMzilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc2lvb19zdGF0ZSI6IjUwNTBkNGJmLTE0NjEtNGU4OC1hzjU2LTv1
Mzd1Y2UzYjY3OCIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjEwMTY4OCJ9.3o
tt7xSmfCc-ZZI5JxvDhUCKEfeg7FmMcTc0NBiIyvQ"
}
}

```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 4. InitConsent

POST [\({{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}\)](https://{{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}})

Params Authorization Headers (12) Body Pre-request Script Tests Settings Cookies

Headers (9 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input type="checkbox"/>					
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 4. InitConsent

POST [\({{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}\)](https://{{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}})

Params Authorization Headers (12) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1
2     "codeBank": "BCR",
3     "periodOfValidity": 90,
4     "psuEmail": "catalin.zaman@itsmartsystems.eu",
5     "psuIntermediaryId": "123",
6     "redirectURL": "https://www.google.ro",
7     "TAccepted": true
8

```

Response:

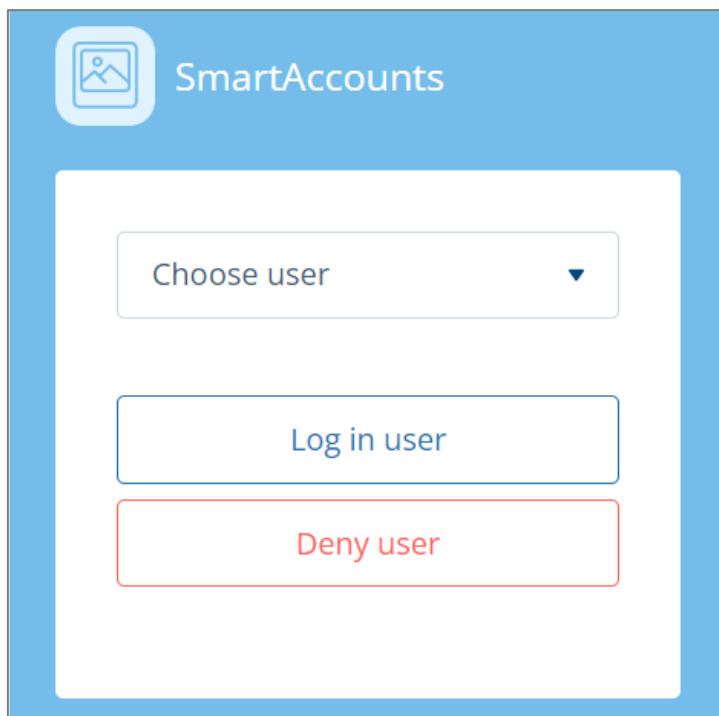
```

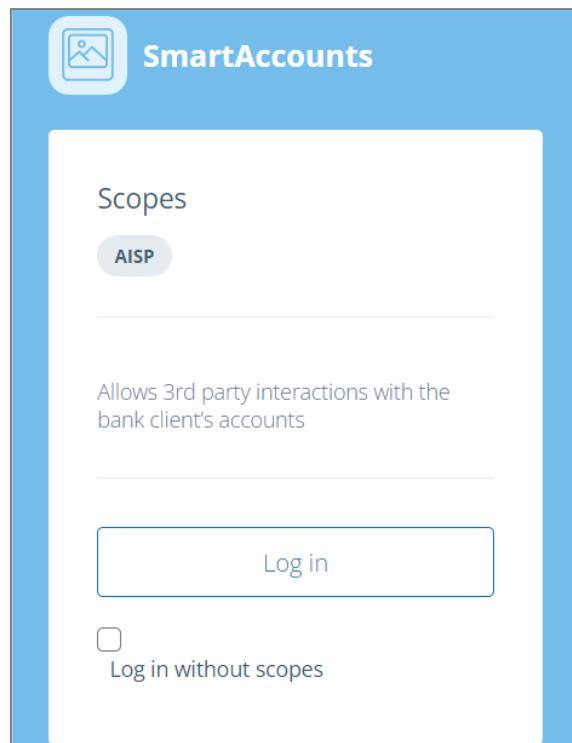
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {

```

```
"scaOAuth": {
    "href": "https://webapi.developers.erstegroup.com/api/bcr/sandbox/v1/sandbox-
idp/auth?state=eyJlbmMiOiJBMTI4Q0JDLUhTMjU2IiwiYWxnIjoiZGlyIiwiY3R5IjoiSldUIIn0..7nG
E09UoFJ5d_L1828va8g._6ErnUP9ELZxVYru8XAByOwrhXYIBriMWdxN7anLyjd8QtREVRztjsAWuzKElsS
kk6b4y5P0OGcc4oHWzz01365JrghPGF-m2-f_wY-q14-
FIJKKbORnUTCtb5qplaJoPMGRk6dJIEBh4tcSCVn--FCn7WdVtQ-
z51I9myvu3tW9sEy7P4kCfb5wHfb5Djm-5fxTkBlS-2_hHravEx5uSzUIorKf9RT1B3hRuYBv3qMaRS-
pZz069XL2BUuDoduR3Zmeh7eKyN_PqH6dQEbz5YBTqo_65-bsMyqe4ds1Q2qB7SjcvrWWXDUqss1_-
TM590i_ae_mmjKrJjjhXvZZMowleZemgnJXN8mdjkJFB127bSUio24a3uf9PAifPnR4pdN1L_6X-
VaSYAGmXoAIE8SXvVpYGgZXT_oAPDVaP15MCxCeCAYDoxtK_vE7jyJwpvtbv5T4kjgU8uug5EQQQKwiv9mQ6h
RLb8tqdDtyanaTkPDeiupAhVhtO5nwsa62n447BPGDxaKLqlwf3lWobR65vqZBYaRu4kQ1XKYTtmp4eEZlh
5h1yq8DEi1iD9gKOGj-zLL4_GjA0z84IxA_Kvr8q9hquNUVBw7sybzqQ9HjYo0-
MIPkczD6WTy7uaLvaQWXJ1hROVu548aNFRqnLAN6pBPjh1snKRI2lTQ1Ut3EsJMzHtrcM8JqcliO2TR5twy
c0irm5O19j1E71AKxBK6dvIwI5kop5QHT3oJfLo2vzvOn84msezc80Iui2vC7AiPjZ1Pu4XrCOSW8huzj_
nMp9eXLUTwbp8SnB2sA4846KrnLUhWsH2e8ERYTYc4LkSckTTa36gj9mN0Q6s0E_KAlqaaAwd-
mNzHeGmN1qFp_oWdcsgjIxelHuxfZ2-kif0nVMhxP_rIGFF28vThZMv-
qOrUZMhxAkgkMYrzab5Sm0wRRi4c-s61yf5VlO-9QTZ2PUGPV2rx15Cs_RMdg6QlilqILONW639KmmHK_W-
JL9xvK9TDBT0EU4QjLW2RQvekJEvUX2mpqwBBHrNSoIUiSUVkQKJQ5c6JttvnyHvY.nsx2R5KPQ6kvbQBj4
SviVw&response_type=code&client_id=abf79a52-9aed-47c6-be5d-
fb6b96e63a13&redirect_uri=https://api.sandboxaccounts.smartfintech.eu/gateway/core/
rest/api/confirmation&scope=AISP&access_type=offline&transactionId=e10e5683-49e1-
44bf-eb6b-1e3b70243ecd"
}
}
```

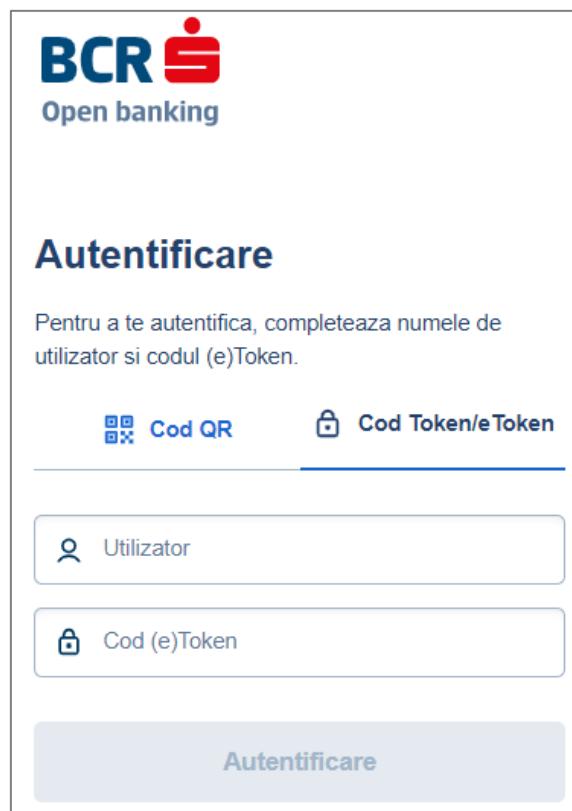
2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application





After pressing "Log in", thus authorising the consent for all available accounts (see information below regarding BCR concept of consent), developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application



After choosing the way of the authentication and pressing "Autentificare", thus authorising the consent for all available accounts (see information below regarding BCR concept of consent), PSU will be redirected to the established page.

3. GetConsent

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
Content-Type	application/json		
PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [],
      "balances": [],
      "transactions": [],
      "availableAccounts": "allAccounts"
    },
    "consentStatus": "Valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-07",
    "validUntil": "2023-05-07",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

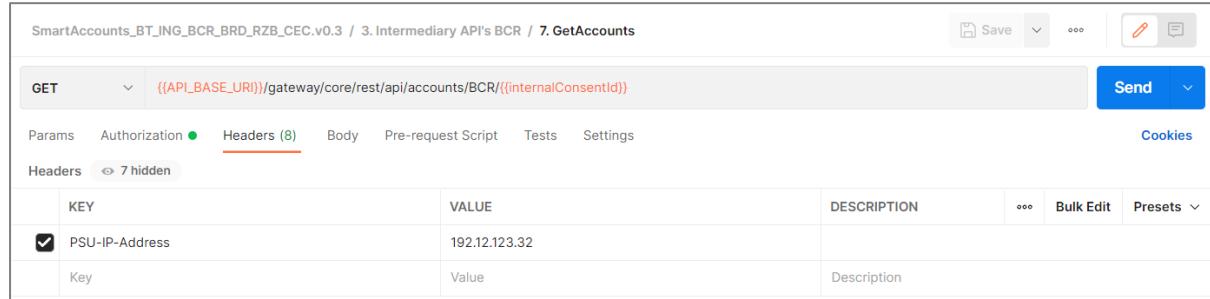
4. GetConsentStatus

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
Content-Type	application/json		
PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "consentStatus": "Valid"  
    }  
}
```

5. GetAccounts



The screenshot shows the Postman interface with the following details:

- Method: GET
- URL: `((API_BASE_URL))/gateway/core/rest/api/accounts/BCR/{internalConsentId}`
- Headers tab selected, showing one header entry:

 - Key: PSU-IP-Address
 - Value: 192.12.123.32

- Other tabs: Params, Authorization, Body, Pre-request Script, Tests, Settings, Cookies.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": [  
        {  
            "resourceId": "11e07590-27d4-409e-914b-c248fb945b11",  
            "iban": "RO77RNCB0002000000010001",  
            "currency": "EUR",  
            "name": "Peter Pan",  
            "product": null,  
            "cashAccountType": "CACC",  
            "status": "enabled",  
            "bic": "RORNCBROB0XXX",  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        },  
        {  
            "resourceId": "62b91e9a-3074-11e9-b210-d663bd873d93",  
            "iban": "RO50RNCB0002000000010002",  
            "currency": "EUR",  
            "name": "My current account",  
            "product": null,  
            "cashAccountType": "CACC",  
            "status": "enabled",  
            "bic": "RORNCBROB0XXX",  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        }  
    ]  
}
```

```

        "product": null,
        "cashAccountType": "CACC",
        "status": "enabled",
        "bic": "RORNCBROB0XXX",
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3' with a selected environment '3. Intermediary API's BCR'. The request is for '8. GetBalances' using a GET method. The URL is `((API_BASE_URI))/gateway/core/rest/api/accounts/BCR/62b91e9a-3074-11e9-b210-d663bd873d93/((internalConsentId))/balances`. The 'Headers' tab is active, showing two entries: 'PSU-IP-Address' with value '192.12.123.32' and 'Key' with value 'Value'. Other tabs include 'Params', 'Authorization', 'Body', 'Pre-request Script', 'Tests', 'Settings', and 'Cookies'. A 'Send' button is visible at the top right.

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO50RNCB0002000000010002"
        },
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 400.0
                },
                "balanceType": "closingBooked",
                "referenceDate": "2017-11-21",
                "creditLimitIncluded": null,
                "creditDebitIndicator": null,
                "lastChangeDateTime": null,
                "lastCommittedTransaction": null,
                "transactionStatus": null
            }
        ]
    }
}
```

```

        ]
    }
}

```

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 9. GetTransactions

GET [{{API_BASE_URI}}/gateway/core/rest/api/accounts/BCR/62b91e9a-3074-11e9-b210-d663bd873d93/{{internalConsentId}}/transactions?bookingStat...](#)

Params ● Authorization ● **Headers (8)** Body Pre-request Script Tests Settings Cookies

Headers 7 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 9. GetTransactions

GET [{{API_BASE_URI}}/gateway/core/rest/api/accounts/BCR/62b91e9a-3074-11e9-b210-d663bd873d93/{{internalConsentId}}/transactions?bookingStat...](#)

Params ● Authorization ● **Headers (8)** Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> bookingStatus	booked			
<input checked="" type="checkbox"/> dateFrom	2022-05-05			
<input checked="" type="checkbox"/> dateTo	2022-05-15			
Key	Value	Description		

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO50RNCB0002000000010002",
      "currency": "EUR"
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "trx3",
          "entryReference": null,
          "endToEndId": "End to End identification3",
          "bookingDate": "2018-10-02",
          "valueDate": "2018-10-02",
          "executionDateTime": null,
          "transactionAmount": {
            "currency": "EUR",
            "amount": -1000.5
          },
          "transactionType": null,
        }
      ]
    }
  }
}
```

```

        "exchangeRate": [],
        "creditorName": "The Hulk",
        "creditorAccount": {
            "iban": "AT108809156711523429"
        },
        "debtorName": "Frodo",
        "debtorAccount": {
            "iban": "RO50RNCB0002000000010002"
        },
        "remittanceInformationUnstructured": "Credit Transfer",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    },
    {
        "transactionId": "W100000000000707",
        "entryReference": null,
        "endToEndId": "19RQDEXNN1S1",
        "bookingDate": "2023-01-11",
        "valueDate": "2023-01-11",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "EUR",
            "amount": -0.01
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": "Rncbro benf",
        "creditorAccount": {
            "iban": "RO23RNCB0002000000010003"
        },
        "debtorName": null,
        "debtorAccount": {
            "iban": "RO50RNCB0002000000010002"
        },
        "remittanceInformationUnstructured": "19RQDEXNM6WY",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
],
"pending": [],
"links": null
},
"balances": []
}
}

```

8. GetTransactionsDetails

The screenshot shows the Postman interface with a GET request to the endpoint `/api/accounts/BCR/62b91e9a-3074-11e9-b210-d663bd873d93/transactions/trx3/details`. The Headers tab is active, displaying the following configuration:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "transactionId": "trx3",
    "entryReference": null,
    "endToEndId": "End to End identification3",
    "bookingDate": "2018-10-02",
    "valueDate": "2018-10-02",
    "transactionAmount": {
      "currency": "EUR",
      "amount": -1000.5
    },
    "transactionType": null,
    "creditorName": "The Hulk",
    "creditorAccount": {
      "iban": "AT108809156711523429"
    },
    "debtorName": "Frodo",
    "debtorAccount": {
      "iban": "RO50RNCB0002000000010002"
    },
    "remittanceInformationUnstructured": "Credit Transfer",
    "executionDateTime": null
  }
}
```

9. GetAccountDetails

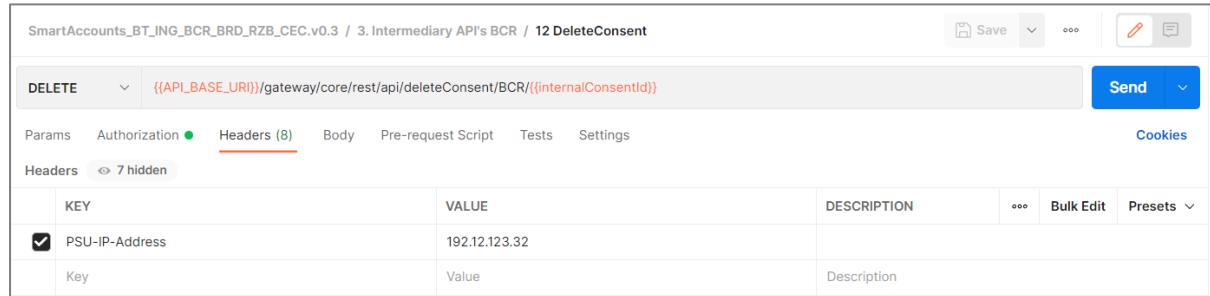
The screenshot shows the Postman interface with a GET request to the endpoint `/api/accounts/BCR/11e07590-27d4-409e-914b-c248fb945b11/details`. The Headers tab is active, displaying the following configuration:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "resourceId": "11e07590-27d4-409e-914b-c248fb945b11",  
        "iban": "RO77RNCB0002000000010001",  
        "currency": "EUR",  
        "name": "Peter Pan",  
        "product": null,  
        "cashAccountType": "CACC",  
        "status": "enabled",  
        "bic": "RORNCBROB0XXX",  
        "accountOfficerDescription": null,  
        "openingDate": null,  
        "customerSector": null,  
        "customerType": null,  
        "accountUsage": null,  
        "workingBalance": null,  
        "onlineActualBalance": null,  
        "balances": [],  
        "sortCodeAccountNumber": null  
    }  
}
```

10. DeleteConsent



The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3' with a selected environment '3.Intermediary API's BCR'. The 'DeleteConsent' request is selected, showing a DELETE method at the URL {{API_BASE_URI}}/gateway/core/rest/api/deleteConsent/BCR/{{internalConsentId}}. The 'Headers' tab is active, displaying two entries: 'PSU-IP-Address' with value '192.12.123.32' and 'Key' with value 'Value'. Other tabs include Params, Authorization, Body, Pre-request Script, Tests, Settings, and Cookies.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "message": "Consent has been successfully revoked"  
    }  
}
```

11. RefreshToken

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Headers (10)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> internalConsentId	{{{internalConsentId}}}				
<input checked="" type="checkbox"/> accessToken	{{{accessToken}}}				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Body

form-data

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	{{{refresh_token}}}			
<input checked="" type="checkbox"/> client_id	{{{client_id-partner}}}			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 3. Intermediary API's BCR / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Tests

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);
    
```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101688,
    "accessToken": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFjMi1NUUF4SEJXYmlxb2ZZJIn0.eyJleHAiOjE2NTUyOTk5NjQsImIhdCI6MTY1NTI5OTY2MywianRpIjoiYjFiMDkwYmEtODNlZi00NDMyLtk5ZWUtODZiNDljNGM1MTExIiwiaxXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11LOWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNTA1MGQ0YmYtMTQ2MS00ZTg4LWFmNTYtNWUzN2VjZTNiNjc4IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2Nlc3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbGllbnR2MSI6eyJyb2xlcyI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjbj3VudCI6eyJyb2xlcyI6WyJtYW5hZ2UtYWNjb3VudCIsImlhbmFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1Ii119fSwic2NvcGUiOjZbWFydGFjY291bnRzX2NvbnNlbnRfaWRfy2xhaW06MTAxNjg4IiwiC21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltIjoiMTAxNjg4IiwiY2xpZW50SWQiOj0ZXX
```

```

0Y2xpZW50djEiLCJjbGllbnRIB3N0IjoiMTAuMC4yNDguMzIiLCJlbWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn
ByZWlcnJlZF91c2VybmrTzSI6InNlcnPbY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbGllbnRBZGRyZ
XNzIjoiMTAuMC4yNDguMzIifQ.Fqu-
7bQd9Ovi9AQFy05qHcdq99oEBb9eRP65SAwcf7oXbnJOzUA8vawrLBW1dnx1Hw2069MwdqtBuhwI-
z6PLDxo8qXXLc9luc10kda_IdDxLHGkGMTv73-f2B2U9Yds4cIW4NthR9-uqf91dJRwpF8UtrQOpKIC1W-
iot29v6zjEzBx_qR-Slnx30CsALHo04u73EqjRPq_US-
5tuKAn6TIWrwgL_ZpkX7Nq9PDgcP2zEfkPnAlbEo4LrGs0XBfx5NFAv9yVsrfZOiOPskcwYDFMgxLcA51N
IrJumpG1eZBzgX9cYrCkj6Eda9Uv0gpdkjSD1GQDX15mXjmJAK2A",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC13OWNhY
2RmMiljk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzU2MjMsImlhdCI6MTY1N
TI5OTY2NCwianRpIjoiOTEwZWU1OGItMze0NS00M21LWExMmItOWQ3ZWM0NDJjMWRIiwiiaXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NTYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFmZi1mOTc4MTU2NDC3ZDMiLCJ0eXAiOjJPZmZsaW5lIiwiY
XpwIjoidGVzdGNsaWVudHYxIwic2Vzc2lvb19zdGF0ZSI6IjUwNTBkNGJmLTE0NjEtNGU4OC1hzjU2LTV1
Mzdly2UzYjY3OCIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjEwMTY4OCJ9.a_
CI0TAecsL_3uoU0POfewbz7p-q17kcjB49eEE1h24"
}
}

```

In case the "accessToken" expires, the response will have the following structure:

```
{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}
```

4. BRD PJ (corporate) - Flow

1. Authenticate

The screenshot shows the Postman interface for making an authentication request. The method is set to POST, and the URL is {{(MTLS_BASE_URI)}}/gateway/authenticate/rest/api/token. In the Headers section, there are 7 items. The Body section is set to form-data, with a single parameter named 'client_id' whose value is {{(client_id-partner)}}. The Params section also contains a 'client_id' parameter with the same value. The 'Send' button is visible at the top right of the request configuration area.

```

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 4. Intermediary API's BRD / 1. Authenticate
POST {{MTLS_BASE_URL}}/gateway/authenticate/rest/api/token
Params Authorization Headers (7) Body • Pre-request Script Tests • Settings • Cookies
1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116504,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzYwMjEyMTEsImlhDCI6MTY3NjAyMDkxMSwianRpIjoiY2FmZGMyYmUtYjVhYi00NTvkLThmNWYtN2ZkNjQ3Y2NhZTcwIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oJgwODAvYXX0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfyzXIIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaWuX3N0YXR1IjoiYzc2NjMxZTEtZTl2MS00NDM5LTkyYjAtNzgzM2YyOTU3MjEyIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFBsbV9hY2N1c3MiOnsicm9sZXMiolsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtYW5hZ2UtYWNjb3VudCIisIm1hbmFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoijzbWFydGFjY291bnRzX2Nvbnn1bnRfaWRfY2xhaW06MTE2NTA0Iiwigc21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltijoimTE2NTA0IiwiY2xpZW50SWQioiJ0ZXN0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xODMuMTQyIiwiZ1haWxfdmVyaWZpZWQiOmZhbHN1LCJwcmVmZXJyZWRfdXN1cm5hbWUiOjzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkmVzcyI6IjEwLjAuMTgzLjE0MiJ9.RAwErprQrWHHCkJ6k5GRuRNHxWmM-VYI48Jhh-wuqfNwMLFWNP88F5f0QTavf30GPkW5ASFWRyOsG2Jjocjonv-FA42VhG7vJc5nPBNQyDKYiEm6-CnbZTW4AC9Prpw9zUg_NjHA9BPXjc66DaIPt1Zghok7qBvRhGaJjk68htRw4DkVidHPtnCqHEdlEeECjK0Hec_5ymnBSzorubBTkYcDMMjhjn2RsujB3zN_m5tsPdPGhjEt-DBGzb09XYuS4Ti6MCKzTEmPOEsRdtNeezmPeWgGaKCW5Eh-Ju-J11Vf88zIjk8k1FTf8pv8p1ynX-XaVvWBBitQMr01q2cbA",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhy2RmMi1kMjk3LTQ0YWItyjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM3OTY5MTEsImlhDCI6MTY3NjAyMDkxMSwianRpIjoiYTZjNjZhMzQtMWjkZi00OTyztLtg5NzItZjMwOT11ZD1kNmExIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oJgwODAvYXX0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiYXpwljoidGVzdGNsaWVudHYxIiwigc2Vzc2lvb19zdGF0ZSI6ImM3NjYzMwUxLWUyNjEtNDQzOS05MmIwLtc4MzNmMjk1NzIxMiIsInNjb3BlIjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjUwNCJ9._ShOG4HBdBNMqm_8X40XDysaexLLSOxBWVb01k1yMzE"
  }
}
```

2. initConsent

The image contains two screenshots of the Postman API testing tool, both titled "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 4. Intermediary API's BRD / 4. initConsent".

Screenshot 1 (Headers Tab):

KEY	VALUE	DESCRIPTION	... Bulk Edit	Presets
Content-Type	application/json			
PSU-IP-Address	192.12.123.32			
PSU-ID	13333330			
PSU-ID-Type	PF			
PSU-ID	31111111			
PSU-ID-Type	PJ			
Key	Value	Description		

Screenshot 2 (Body Tab):

Body Type: JSON

```

1
2   ...
3     "codeBank": "BRD",
4     "periodOfValidity": 90,
5     "psuEmail": "bogdan.gurita@itsmartsystems.eu",
6     "psuIntermediaryId": "123",
7     "redirectURL": "https://google.com",
8     "TCaccepted": true
  
```

The user needs to have "PSU-ID" and "PSU-ID-Type" checked on headers ("PSU-ID-Type" = "PJ" in this case).

By being decoupled, the user the authentication of the PSU occurs outside of the interaction with the intermediary page (the PSU must log in in the internet banking application to authorise the consent).

The PSU should know which internet banking application should be used to authorise the consent.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "links": {
      "scaOAuth": {
        "href": null
      }
    }
  }
}
```

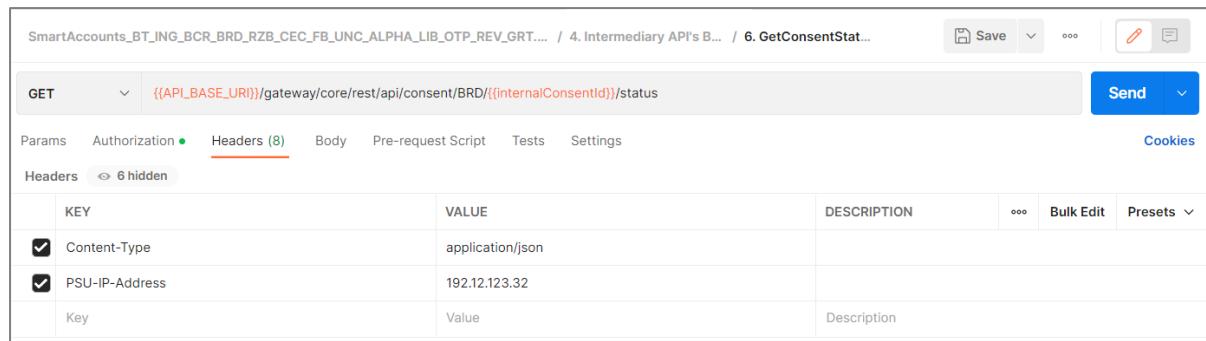
3. GetConsent

The screenshot shows the Postman application interface. A GET request is being prepared to the URL `((API_BASE_URI))/gateway/core/rest/api/consent/BRD/({internalConsentId})`. The 'Headers' tab is active, displaying two header entries: 'PSU-IP-Address' with value '192.12.123.32' and 'Content-Type' with value 'application/json'. Other tabs like 'Params', 'Authorization', 'Body', etc., are visible but not selected.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO11BRDE111SV111111111111"
        }
      ],
      "balances": [
        {
          "iban": "RO11BRDE111SV111111111111"
        },
        {
          "iban": "RO11BRDE111SV111111112222",
          "currency": "RON"
        },
        {
          "iban": "RO22BRDE222SV222222222222"
        }
      ],
      "transactions": [
        {
          "iban": "RO11BRDE111SV111111111111"
        }
      ],
      "availableAccounts": null
    },
    "consentStatus": "received",
    "frequencyPerDay": 4,
    "lastActionDate": "2019-03-01",
    "validUntil": "2019-05-01",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

4. GetConsentStatus



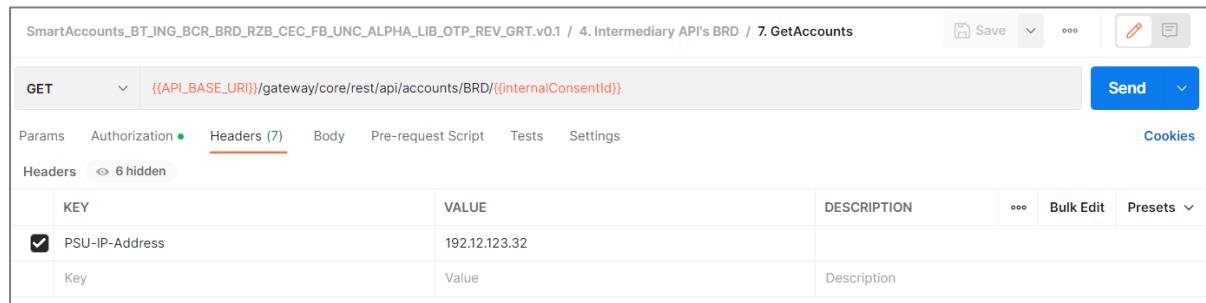
The screenshot shows the Postman interface with a GET request to `{API_BASE_URI}}/gateway/core/rest/api/consent/BRD/{internalConsentId}/status`. The Headers tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
Content-Type	application/json				
PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": {  
    "consentStatus": "received"  
  }  
}
```

5. GetAccounts



The screenshot shows the Postman interface with a GET request to `{API_BASE_URI}}/gateway/core/rest/api/accounts/BRD/{internalConsentId}`. The Headers tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": [  
    {  
      "resourceId": "46MS",  
      "iban": "RO11BRDE111SV111111111111",  
      "currency": "EUR",  
      "name": null,  
      "product": null,  
      "cashAccountType": null,  
      "status": null,  
      "bic": null,  
      "accountOfficerDescription": null,  
      "openingDate": null,  
      "customerSector": null,  
      "customerType": null,  
      "branch": null  
    }  
  ]  
}
```

```

        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    },
    {
        "resourceId": "46NB",
        "iban": "RO22BRDE222SV2222222222",
        "currency": "RON",
        "name": null,
        "product": null,
        "cashAccountType": null,
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/BRD/RO47BRDE410SV04747474100/{{internalConsentId}}/balances
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Buttons:** Save, Send

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {},
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": -229959.14
                }
            }
        ]
    }
}
```

```
        },
        "balanceType": "",
        "referenceDate": "2013-05-26",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null,
        "lastChangeDateTime": null,
        "lastCommittedTransaction": null,
        "transactionStatus": null
    }
]
}
```

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB_OTP_REV_GRT.v... / 4. Intermediary API's BR... / 9. GetTransactio...

Save ⌂ ⌄ ⌅ ⌆

GET {{API_BASE_URL}}/gateway/core/rest/api/accounts/BRD/101000000195/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-

Send

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> bookingStatus	booked			
<input checked="" type="checkbox"/> dateFrom	2023-01-01			
<input checked="" type="checkbox"/> dateTo	2023-02-10			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 4. Intermediary API's BR... / 9. GetTransactio...

Save ⚙️

Send

GET {{API_BASE_URL}}/gateway/core/rest/api/accounts/BRD/1010000000195/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Headers • 6 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> next	{{next_page}}				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 4. Intermediary API's BR... / 9. GetTransactio...

Save  

Send 

GET  {{API_BASE_URI}}/gateway/core/rest/api/accounts/BRD/1010000000195/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-

Params • Authorization • Headers (8) Body Pre-request Script Tests  Settings Cookies

```
1 | const responseJson = pm.response.json();
2 | let next = responseJson.payload.transactions.links;
3 | if (next != null) {
4 |   postman.setEnvironmentVariable("next_page", next.next.href);
5 | }
6 | else {
7 |   postman.setEnvironmentVariable("next_page", null);
8 | }
```

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",
```

```

"payload": {
    "account": {
        "iban": "RO12BRDE12SV123455601001",
        "currency": "RON"
    },
    "transactions": {
        "booked": [
            {
                "transactionId": null,
                "entryReference": null,
                "endToEndId": null,
                "bookingDate": "2022-08-01",
                "valueDate": "2022-08-01",
                "executionDateTime": null,
                "transactionAmount": {
                    "currency": "RON",
                    "amount": -1275.01
                },
                "transactionType": null,
                "exchangeRate": [],
                "creditorName": null,
                "creditorAccount": {
                    "iban": "RO76BRDE010SV12345710100"
                },
                "debtorName": null,
                "debtorAccount": {
                    "iban": "RO12BRDE12SV123455601001"
                },
                "remittanceInformationUnstructured": "TRFOPH2022081114453,Tr.Credit-Plata,303011008,BENEFICIAR PLATA,CUI/CNP1234567,DETALII-PLATA-DETALII-PLATA,-DETALII",
                "proprietaryBankTransactionCode": "Tr.Credit-Plata",
                "additionalInformation": null,
                "smartRemittance": null,
                "transactionStatus": null,
                "creditDebitIndicator": null
            },
            {
                "transactionId": null,
                "entryReference": null,
                "endToEndId": null,
                "bookingDate": "2022-08-02",
                "valueDate": "2022-08-02",
                "executionDateTime": null,
                "transactionAmount": {
                    "currency": "RON",
                    "amount": -10000.0
                },
                "transactionType": null,
                "exchangeRate": [],
                "creditorName": null,

```

```

        "creditorAccount": {
            "iban": null
        },
        "debtorName": null,
        "debtorAccount": {
            "iban": "RO12BRDE12SV123455601001"
        },
        "remittanceInformationUnstructured": "TRFOP433695436645465656, U
tiliz POS B pl,MAGAZIN ALBA-
IULIA,Card nr 427398*****6523 (V,BD) (DETINATOR CARD) MAGAZ,IN ALBA-IULIA",
        "proprietaryBankTransactionCode": "Utiliz POS B pl",
        "additionalInformation": null,
        "smartRemittance": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
],
"pending": [],
"links": null
},
"balances": []
}
}

```

8. GetAccountDetails

The screenshot shows the Postman interface with a GET request to `/accounts/BRD/RO47BRDE410SV04747474100/({internalConsentId})/details`. The Headers tab is active, displaying the following entries:

KEY	VALUE	DESCRIPTION
PSU-IP-Address	192.12.123.32	
Key	Value	Description

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "resourceId": "46MS",
        "iban": "RO11BRDE111SV111111111111",
        "currency": "EUR",
        "name": null,
        "product": null,
        "cashAccountType": null,
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "closingDate": null
    }
}
```

```

        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}

```

9. DeleteConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 4. Intermediary API's BRD / 11 DeleteConsent

DELETE [\({{API_BASE_URI}}/gateway/core/rest/api/deleteConsent/BRD/{{internalConsentId}}\)](https://{{API_BASE_URI}}/gateway/core/rest/api/deleteConsent/BRD/{{internalConsentId}})

Send

Params	Authorization	Headers (7)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers (6 hidden)								
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	PSU-IP-Address	192.12.123.32						
	Key	Value			Description			

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "message": "Consent has been successfully revoked"
    }
}

```

10. RefreshToken

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 4. Intermediary API's BRD / 1.1 RefreshToken

POST [\({{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken\)](https://{{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken)

Send

Params	Authorization	Headers (9)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers (7 hidden)								
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	internalConsentId		{{{internalConsentId}}}					
<input checked="" type="checkbox"/>	access_token		{{{access_token}}}					
	Key	Value			Description			

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116504,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFIMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzYwMjIwOTQsImhdCI6MTY3NjAyMTc5NCwianRpIjoiMDIyMDFjMzMtY2U3My00YTQ5LWEzYmMtMzU5ZTRiNTFjMjEzIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfyzXIIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYzc2NjMxZTEtZT12MS00NDM5LTkyYjAtNzgzM2YyOTU3MjEyIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjdLCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InR1c3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYWN5hz2UtYWNjb3VudCIsIm1hbmcFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnN1bnRfaWRFy2xhaW06MTE2NTA0IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2NTA0IiwiY2xpZW50SWQiOj0ZGN0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xODMuMTQyIiwiZW1haWxfdmVyaWZpZWQiOmZhbHN1LCJwcmVmZXJyZWRfdXN1cm5hbwUiOjzZXJ2aN1LWFjY291bnQtdGVzdGNSaWVudHxIiwiY2xpZW50QWRkmVzcyI6IjEwLjAuMTqzLjE0MiJ9.GQhSxw9FyIncN6_iF9vEkxHVOBtxYMFfeeivi8VGC9sIC_QN717NGOpVsOc2cwpqP3ZFM63z_Mrd6yy3IZK0mCgCtwP_LtQ3JqdUIrml6q21nuU1tfbLUeQPdElY5GvAo1V3znIJbDAtOE7tKTcCCdu1_hcSy-WX8vevNlfssaa-bV8HwvG161NiBYCdHoLcgq6WJj3-RbkDc1a8GyG0BpP0vwxE49GXxPtsnQC0cN1cVBqAE04JroqgrgInajI6pA6FU30RPdcqfShjogOViIdE0C3w0uTa5sTh7A1h1zOibfUCtdR81bfpxYN6UuQBrm6LDYBysbXFjcfRW7LUyjwVQ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS2RmMi1kMjk3LTQ0YWItYjY5Ni05NtC0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM3OTY5MTEsImhdCI6MTY3NjAyMTc5NCwianRpIjoiNTIxZjAzMzEtM2ZmOC00NDE4LTg3NzMtODUxMGY5ODI50WU4IiwiA XNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLCJzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY"
  }
}
```

```

XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc2lvb19zdGF0ZSI6ImM3NjYzMWUxLWUYNjEtNDQzOS05MmIwLTc4
MzNmMjk1NzIxMiIsInNjb3BlIjoic21hcncRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjExNjUwNCJ9.aJ
3ESKGU_vfrUzlIm1jY6vqjeDD6OpCHz172UjYjyow"
}
}

```

In case the "accessToken" expires, the response will have the following structure:

```
{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}
```

5. BRD PF (individuals) - Flow

1. Authenticate

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> client_id	<code>{{{client_id-partner}}}</code>			
Key	Value	Description		

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101672,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldeIiwi21kIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTM4OTUsImlhd
CI6MTY1NTI5MzU5NSwianRpIjoizTE0OWIzNTItYTU3NC00M2JiLTgwMDktMzI5MDIwZjc5OTUwIiwiAXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcncRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZT1yMC11LOWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfYzxIiLCJhenAiOj0ZxN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYzg2MWNhZDEtN
mVhOS00ZTBkLWE0MGYtZWM5Yzc2MGYwMzExIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFBsV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdG1vbiJ
dfSwicmVzb3VyY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdG1vbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtyW5hZ2UTYWNjb3VudCIIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1Ii119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbNlbnRfaWRfY2xhaW06MTAx
NjcyIiwiC21hcncRhY2NvdW50c19jb25zzW50X21kX2NsYWltIjoiMTAxNjcyIiwiY2xpZW50SWQiOj0ZxN
0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn
```

```

ByZWZlcnJlZF91c2VybmFtZSI6InNlcnPjY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbGllbnRBZGRyZ
XNzIjoiMTAuMC4yNDguMzIifQ.BDf-
7K7V_nTlBilj5pWkcRFwBw69KZZOST1XVUNNjGGmGfGinUUTpfmJMqCbtLaNTURhYzrSYDlkcuTJ7oKk1f
4MQpKR7W3GvEwJnoJ7Pp6fw1R17TewMQ9QBAGgB1xfk8iAVx0CiRv9Tvj_IvmkgTAI-
XB03tyrFZQn0xbg7kWmVFJx_iG-g5W93qCGETynoVEnMJEEdqsUDXLi6xa3_QATcF8L8FN0v-
NVX9TvYorEC_XJ16LmkItkhiBK_6klvyZSV9R2kjE_nkqSbio9kv31KWG9Wg-
NuK2pwiqE1feQ_VIRpKm_H8eDDCcF1L1FFVviuOdXbZQY59E7g-bBQQ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldeIiA6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNjk1OTUsImhdCI6MTY1N
TI5MzU5NSwianRpIjoiMmNhMGUyZjUtMzY2My00NzkwLWF1NmEtODlmNjM0MmE2ZDVjIiwiaXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhy2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWVmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOiJPZmZsaW5lIiwiY
XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc21vb19zdGF0ZSI6ImM4NjFjYWQxLTZlYTktNGUwZC1hNDBmLWVj
OWM3NjBmMDMxMSIsInNjb3B1Ijoic21hcRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjEwMTY3MiJ9.xk
NqUCnOezXS1ffZdbmcDN5CT97ubfoXg4fMlwQN0Hk"
}
}

```

2. initConsent

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> PSU-ID	13333330				
<input checked="" type="checkbox"/> PSU-ID-Type	PF				
<input type="checkbox"/> PSU-ID	31111111				
<input type="checkbox"/> PSU-ID-Type	PJ				

```

1   "codeBank": "BRD",
2   "periodOfValidity": 90,
3   "psuEmail": "monica.rotaru@itsmartsystems.eu",
4   "psuIntermediaId": "123",
5   "redirectURL": "https://google.com",
6   "TCAccepted": true
7
8

```

Response:

```

{
  "status": 200,
  "messageStatus": "Success",
}

```

```

"payload": {
    "consentStatus": "RECEIVED",
    "consentId": "800000022",
    "_links": {
        "scaOAuth": {
            "href": null
        }
    }
}
}

```

The user needs to have "PSU-ID" and "PSU-ID-Type" checked on headers ("PSU-ID-Type" = "PF" in this case).

By being decoupled, the user the authentication of the PSU occurs outside of the interaction with the intermediary page (the PSU must log in in the internet banking application to authorise the consent).

The PSU should know which internet banking application should be used to authorise the consent.

3. GET Consent

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3'. The '4. GetConsent' request is selected. The method is 'GET' and the URL is `((API_BASE_URI))/gateway/core/rest/api/consent/BRD/{{internalConsentId}}`. The 'Headers' tab is active, showing the following configuration:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
<input checked="" type="checkbox"/> Content-Type	application/json			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "access": {
            "accounts": [],
            "balances": [],
            "transactions": [],
            "availableAccounts": "allAccounts"
        },
        "consentStatus": "received",
        "frequencyPerDay": 4,
        "lastActionDate": null,
        "validUntil": "2019-09-14",
        "recurringIndicator": true,
        "combinedServiceIndicator": null,
    }
}
```

```

        "creationDate": null
    }
}

```

4. GET ConsentStatus

The screenshot shows the Postman interface with a GET request to `({{API_BASE_URI}})/gateway/core/rest/api/consent/BRD/{{internalConsentId}}/status`. The Headers tab is selected, displaying the following configuration:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json			
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "received"
    }
}
```

5. GET Accounts

The screenshot shows the Postman interface with a GET request to `({{API_BASE_URI}})/gateway/core/rest/api/accounts/BRD/{{internalConsentId}}`. The Headers tab is selected, displaying the following configuration:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": [
        {
            "resourceId": "1010000000195",
            "iban": "RO47BRDE410SV04747474100",
            "currency": "RON",
            "name": "Contul meu in RON",
            "product": null,
            "cashAccountType": null,
            "status": "Activ",
            "id": "1010000000195"
        }
    ]
}
```

```

        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    },
    {
        "resourceId": "1010000000285",
        "iban": "RO46BRDE410SV04646464100",
        "currency": "EUR",
        "name": "Contul meu in EUR",
        "product": null,
        "cashAccountType": null,
        "status": "Activ",
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GET Balances

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3' with a specific request for '4. Intermediary API's BRD / 7.GetBalances'. The request method is 'GET' and the URL is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/BRD/RO47BRDE410SV04747474100/{{internalConsentId}}/balances`. The 'Headers' tab is selected, showing the following header:

KEY	VALUE	DESCRIPTION	... Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.168.1.123			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO47BRDE410SV04747474100",

```

```

        "currency": "RON"
    },
    "balances": [
        {
            "balanceAmount": {
                "currency": "RON",
                "amount": 2020.55
            },
            "balanceType": "expected",
            "referenceDate": null,
            "creditLimitIncluded": "true",
            "creditDebitIndicator": null,
            "lastChangeDateTime": "2019-01-12 03:32:07",
            "lastCommittedTransaction": null,
            "transactionStatus": null
        }
    ]
}
}

```

7. GET Transactions

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** {{API_BASE_URL}}/gateway/core/rest/api/accounts/BRD/101000000195/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2022-
- Params:** bookingStatus (checked), dateFrom (checked), dateTo (checked)
- Query Params:** Key, Value
- Headers:** (9 hidden)
- Body:** (raw, JSON, XML, etc.)
- Tests:** (script)
- Settings:** (cookies, auth, etc.)
- Send:** (button)

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** {{API_BASE_URL}}/gateway/core/rest/api/accounts/BRD/101000000195/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2022-
- Params:** (disabled)
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - next: {{next_page}}
- Body:** (raw, JSON, XML, etc.)
- Tests:** (script)
- Settings:** (cookies, auth, etc.)
- Send:** (button)

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {

```

```

"account": {
    "iban": "RO47BRDE410SV04747474100",
    "currency": "RON"
},
"transactions": {
    "booked": [
        {
            "transactionId": null,
            "entryReference": null,
            "endToEndId": null,
            "bookingDate": "2019-01-13",
            "valueDate": "2019-01-12",
            "executionDateTime": null,
            "transactionAmount": {
                "currency": "EUR",
                "amount": -5.67
            },
            "transactionType": null,
            "exchangeRate": [
                {
                    "currencyFrom": "RON",
                    "rateFrom": 4.7234,
                    "currencyTo": "EUR",
                    "rateDate": "2019-01-13"
                }
            ],
            "creditorName": null,
            "creditorAccount": {
                "iban": "RO46BRDE410SV04646464100"
            },
            "debtorName": null,
            "debtorAccount": {
                "iban": null
            },
            "remittanceInformationUnstructured": "TEST 1",
            "proprietaryBankTransactionCode": "Plata pentru alta sucursala
BRD",
            "additionalInformation": null,
            "transactionStatus": null,
            "creditDebitIndicator": null
        },
        {
            "transactionId": null,
            "entryReference": null,
            "endToEndId": null,
            "bookingDate": "2019-01-12",
            "valueDate": "2019-01-12",
            "executionDateTime": null,
            "transactionAmount": {
                "currency": "RON",
                "amount": 343.01
            }
        }
    ]
}

```

```

        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": null,
        "creditorAccount": {
            "iban": null
        },
        "debtorName": "NUME PRENUME TEST",
        "debtorAccount": {
            "iban": "RO48BRDE410SV04848484100"
        },
        "remittanceInformationUnstructured": "TEST 2",
        "proprietaryBankTransactionCode": "Incasare aceeasi sucursala B
RD",
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
],
"pending": [],
"links": {
    "next": {
        "href": "aHR0cHM6Ly9yZ3cuNWRmNC1iTQ0YTY1ZC51ds1kZS5hcGljb25uZW
N0LmFwcGRvbWFpbj5jbG91ZC9zbWFydGZpbnR1Y2gvc2FuZGJveC9wc2QyLWFpc3AvdjEvYWNjb3VudHMvQ
1JELzEwMTAwMDAwMDAxOTUvdHJhbnNhY3Rpb25zP2RhGvgcm9tPTIwMjMtMDEtMDEMZGF0ZVRvPTIwMjMt
MDItMDgmcVjb3Jkc1BlclBhZ2U9MjAwJmJvb2tpbmdTdGF0dXM9Ym9va2VkJndpdGhCYWxhbmNlPXRYdWU
="
    }
},
"balances": []
}
}

```

8. GET Accounts Details

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 4. Intermediary API's BRD / 10. GetAccountDetails'. The request method is 'GET' and the URL is `{{API_BASE_URI}}/gateway/core/rest/api/accounts/BRD/RO47BRDE410SV04747474100/{{internalConsentId}}/details`. The 'Headers' tab is selected, showing the following header:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {

```

```

        "resourceId": "1010000000195",
        "iban": "RO75BRDE380SV01999963800",
        "currency": "RON",
        "name": "account alias 2",
        "product": null,
        "cashAccountType": null,
        "status": "enabled",
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}
}

```

9. Delete Consent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 4. Intermediary API's BRD / 11 DeleteConsent

DELETE `({{API_BASE_URI}})/gateway/core/rest/api/deleteConsent/BRD/{{internalConsentId}}`

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers	7 hidden							
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	PSU-IP-Address		192.12.123.32					
	Key		Value		Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "message": "Consent has been successfully revoked"
    }
}
```

10. RefreshToken

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 4. Intermediary API's BRD / 1.1 RefreshToken

POST `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/refreshToken`

Send

Params	Authorization	Headers (10)	Body	Pre-request Script	Tests	Settings	Cookies	
Headers	8 hidden							
		KEY	VALUE		DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	internalConsentId		{{internalConsentId}}					
<input checked="" type="checkbox"/>	access_token		{{access_token}}					
	Key		Value		Description			

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	<code>{{refresh_token}}</code>			
<input checked="" type="checkbox"/> client_id	<code>{{client_id-partner}}</code>			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 101673,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZZJIn0.eyJleHAiOjE2NTUyOTYxMjUsImlhICI6MTY1NTI5NTgyNSwianRpIjoiYTZhNza2YWYtNDF1NC00ODYyLWEyNTItYWfhZTrkNTA3ZmUxIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oJgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfyzXiiLCJhenAiOj0ZxN0Y2xpZW50djEiLCJzZXNzaWuX3N0YXR1IjoiNTI5NTdmOWQtN2FjZi00NGQ2LTgwNWQtZju3NGVmYwZmYmY2IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjDLCJyZWFBsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtyW5hZ2UtYWNjb3VudCIisImlhbmFnZS1hY2NvdW50LWxpbtzIiwidm11dy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydGFjY291bnRzX2NvbnnlbnRfaWRfy2xhaW06MTAxNjczIiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYwltIjoiMTAxNjczIiwiY2xpZW50SWQioj0ZxN0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsInByZW1cnJ1ZF91c2VybmtZSI6InN1cnZpY2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZXNzIjoiMTAuMC4yNDguMzIifQ.IzDJOvfRzjelzt44x-jkdS9YR0yKac9giAi0oUto4gcQdBTAKmElCGEFpzvdE7uYH-uToh1sB4PpLyPBSjG2p1Th6pbs9c6HnnfWedtIhqYqNGoQA4bJws0QCQuNxbvrzVcHRNptmOo3o4F2S_ZgK86L9-
oBm6qxayCIHKk9ZI70P8SOQBXPc16_klpeGcFCPY8jm0HzHRLudfCk2eoY_izzMg2xOfkRym34FxoUWXYRe_GSFSbPMPAT5mu-
aMyL_ZgQw_hknhbef1Hun0AXeOsawyijQ5jAh_tFee0Bo27kSxw8PhVZS69jONJ3D6rH6oudDWyix73AGDa_XTzrfwA",
        "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhY2RmM1kMjk3LTQ0YWItYjy5Ni05NtC00DRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzE0MTMsImlhICI6MTY1NTI5NTgyNSwianRpIjoiNGE1MWY4ZGYtNTc0MC00ODV1LWEzYTEtMGZmMTg0NzcxNzdmIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oJgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6Imh0dHA6Ly9rZX1jbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLCJzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiYXpWIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6IjuyOTU3Zj1kLTdhY2YtNDRkNi04MDVklWY1NzR1ZmFmZmJmNiIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYwltOjEwMTY3MyJ9.WG3odBXhY0vJF0LgXj2JHax0WCkk3ZPMu2GYrhOEm3g"
    }
}
```

In case the "accessToken" expires, the response will have the following structure:

```
{
    "status": 401,
    "messageStatus": "Unauthorized",
    "moreDetails": {
        "text": "Invalid access token"
    }
}
```

6. Raiffeisen Bank - Flow

1. Authenticate

The screenshot shows a Postman interface with the following details:

- Method:** POST
- URL:** {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/token
- Body:** (radio button selected)
- Params:** none
- Authorization:** none
- Headers:** (8 items)
- Tests:** (green dot)
- Settings:** (green dot)
- Cookies:** (blue link)

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> client_id	{{{client_id-partner}}}			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 101674,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldeIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFimI1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTYzMTUsImIhdCI6MTY1NTI5NjAxNSwianRpIjoiZDBmMGQ4OGQtOGFmZS00MzI5LWIxZGETM2J1YWExNjJjZWE0IiwiAXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aNWn1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMz1m0Tc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfyzXIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNGQwYjZmYjUtMjVjNC00MTlkLTlmODYtYjU4Zjg1NWJjZDdkIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCIsIm1hbmcFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAxNjc0IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjc0IiwiY2xpZW50SWQiOj0ZXN0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZmllZCI6ZmFsc2UsInByZWZlcnJlZF91c2VybmFtZSI6InNlcnP2Y2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZXNzIjoiMTAuMC4yNDguMzIifQ.gDBj4ay2eDbPeuxkoIKaLNx-wn1R9xg2KJJGPWVzPhxoXANcZMVzNrxJpMw0xsLnDFCnxNCPpxoRmKor9gDA3o8T11Ewo3C4H1HO1is_OQ29QU5rNhBnFdVN2SE0CuBvsW2XPw41T6pE_u80309RM9FFLS0nmcazTntsxBQfWYbu0DzEPTT2Azee6TXiMeZggQCQek90UdI1-1JVBeansOfJaSNZ2I2mze805XHHNRO5z53Dxi90td_iWq2lQvIVqpxzo45gqBSFBpFrriqLK37GModkVz9G0Pj_6-CghGvRNi6-hb9MaiH-giYY1vFI2hnXKwu3SItUQSpW44SMg",
    "refreshToken": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldeIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFimI1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTYzMTUsImIhdCI6MTY1NTI5NjAxNSwianRpIjoiZDBmMGQ4OGQtOGFmZS00MzI5LWIxZGETM2J1YWExNjJjZWE0IiwiAXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aNWn1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMz1m0Tc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfyzXIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNGQwYjZmYjUtMjVjNC00MTlkLTlmODYtYjU4Zjg1NWJjZDdkIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCIsIm1hbmcFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAxNjc0IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjc0IiwiY2xpZW50SWQiOj0ZXN0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZmllZCI6ZmFsc2UsInByZWZlcnJlZF91c2VybmFtZSI6InNlcnP2Y2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZXNzIjoiMTAuMC4yNDguMzIifQ.gDBj4ay2eDbPeuxkoIKaLNx-wn1R9xg2KJJGPWVzPhxoXANcZMVzNrxJpMw0xsLnDFCnxNCPpxoRmKor9gDA3o8T11Ewo3C4H1HO1is_OQ29QU5rNhBnFdVN2SE0CuBvsW2XPw41T6pE_u80309RM9FFLS0nmcazTntsxBQfWYbu0DzEPTT2Azee6TXiMeZggQCQek90UdI1-1JVBeansOfJaSNZ2I2mze805XHHNRO5z53Dxi90td_iWq2lQvIVqpxzo45gqBSFBpFrriqLK37GModkVz9G0Pj_6-CghGvRNi6-hb9MaiH-giYY1vFI2hnXKwu3SItUQSpW44SMg",
    "tokenType": "Bearer"
}
```

```

    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl3OWNhY
2RmI1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzIwMTUsImlh
dCI6MTY1NTI5NjAxNSwianRpIjoINDcyMDViNDgtZDM5Yi00N2JiLWJjNDktNjI4OWQ2MGNiZTViIi
wiaXNzIjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFSbXMvc21hc
nRhY2NvdW50cyIsImF1ZCI6Imh0dHA6Ly9rZX1jbG9hay1hcHAtc2VydmIjZTo4MDgwL2F1dGgvc
mVhbG1zL3NtYXJ0YWNjb3VudHMiLCJzdiOjI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZilmOTc4
MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW5lIiwiYXpWIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb
l9zdGF0ZSI6IjRkMGI2ZmI1LTi1YzQtNDE5ZC05Zjg2lWI1OGY4NTViY2Q3ZCIsInNjb3BlIjoi
c21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY3NCJ9.Jc8ZlCBOHCZRZIMw5c-uVQ
liDyXa6Cd6YRos-muMIBM"
}
}

```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 5. Intermediary API's RZB / **initConsent**

POST `((API_BASE_URI))/gateway/core/rest/api/initConsent/{{internalConsentId}}`

Params Authorization Headers (12) Body Pre-request Script Tests Settings Cookies

Headers (9 hidden)

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json			
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32			
<input checked="" type="checkbox"/> PSU-ID	9999999997			
Key	Value	Description		

NOTE: when initiating a consent for corporate account, then send on API's header the attribute "PSU-ID-TYPE" with value "PJ"

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 5. Intermediary API's RZB / **initConsent**

POST `((API_BASE_URI))/gateway/core/rest/api/initConsent/{{internalConsentId}}`

Params Authorization Headers (12) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1 {
2   ...
3   "allAccounts": [
4     ...
5     ...
6     ...
7     ],
8     ...
9     "codeBank": "RZB",
10    ...
11    "periodofValid": 90,
12    ...
13    "psuEmail": "monica.rotaru@itsmartsystems.eu",
14    ...
15    "psuIntermediarID": "123",
16    ...
17    "redirectURL": "https://google.com",
18    ...
19    "TcAccepted": true
20 }

```

The user needs to have PSU-ID checked and send an IBAN on the request.

Response:

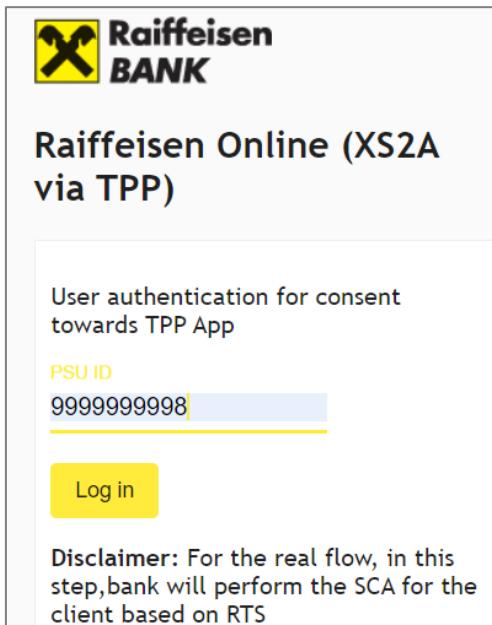
```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    ...
  }
}
```

```

    "consentId": null,
    "_links": {
        "scaOAuth": {
            "href": "https://api-auth-test.raiffeisen.ro/psd2-rbro-oauth2-
api/oauth2/authorize?state=Zx1KbGJtTw1PaUpCTVRJNFEwSkRMVWhUTWpVMklpd2lZV3huSWpvaVpH
bH1JaXdpWTNSNU1qb21TbGRVSW4wLi5CcWlqYVdRbDdvQVk0TENwdWU0ZGznLmRINzRuZ2ZmZlhFYVF1bU1
QS1zOTUh4Z0UyWnJUWDFCNXZCUFZJQ0J6N1zfMkNxTeTJRnjZjQzVqc1JVME1uWWVFcGRDbE5zTGdpdUp1WX
EweG9oY0x0Mzh1TTdyQ1BDaFpUVDJPSkzfQjBndVdZQklsNjZnRVJFbVUxbGhCWGh4M2hDUTVHTVdubV9hM
3Q4Z19mbvUweldLZV9mOFNpZmx1N0JwZV1zdEFsZ1k4V0hDMXA5NUpkNG1pdTkycTFWQ19EYWtZRVczbkxT
YnV2SE1zOVB2MDFvNUJtWnU5ZDBoMk1SaGVBZ05tQ0ZiRWxTT05oMWQ0Qm4tS3FLaFF1WF1vSnU2Ni14Mm5
5dVlwSTZuWmZpWDRmbDM5Mn1xXz1WaEtmWE8tS1hLYURBTjFTd0Z4VU91aXc1SFQyS3IxMG50MEJKMxv6ZF
dZOHzoQ2Rbd3E2cWVhcWJpZGpPVkxHcW93Q19JUVVOTUF2cHVJTHQ1VFE2cTzis3pYdFFIRVVMc1Jyell1XQ
WxrTj16NkJsRHQ0cWdSZXRoaS16TGZjCWNzSUR5TmtkanVQM3ZkT1Q1SV91QV9WMURmbGZoEtLSD1Bej13
M1hmTEhFcDJycjJuN2owOFZ2OFhGOGQ1bGMxMDdYYXJMSVNjRVVHd1hYYzNYVmZKM0M1Z19oWVfkUUpbt0V
VQVZQWU5DM3dDcTRZTE5xUVVTNnRfd3d6REdNaUtZY2d2RXA2MUFXdVNrVmJxQ1FIQm5iYzQ5Q2ItMGhfcl
vodGlxYkd1d0N2OU1hU2QtSFZTaGd5UnowWjJMT1ZCST1PYU1qaGNsZVBXY0c4VE5tZTdHWThrU2ZVSnNQb
jU4cVUTURIUFYyWF10WFdbNkVra2tnVzJpeDh3REpfccjU2RWkxOFEwYS12eE0wcFl4Z1JfWHV4WF1BT1ky
emd5M0NkbVFsc3ZxWGJaNWt3bnhsUUtGV1dtSU9Lcj10Z2djczNMdjJfNGdoZGVITTVsbm1zYV9ueHVR1d
UQ1Z6c0hWR2Y2eUptZEhZem1ITnR1R0R2dEpUdXMzTzdIWUE1ZkN0WFdiWWozci16WFJfenBsS1JKT01pd1
ZwUVBTtmtZY0hhSUppeVZKcjJwdmNMc3ZRTVBEZ0FYRGtaakVxeW5SanpDbG55aUpjN0tSQ1VYS19YM11TX
1RkMW54ekFsYzFneFNCbTkzUGw3d19mTi1ldHNhMktmUkQtV2RSX25VRmF5M2d4ckdrUEFZcE5EU3ZWV1NE
bn1KSUNnQnFFUTJ6OVVCLVZHcnAyWnV1SUVTYzB00EVMbldNT0s0N1RQVUZFU0x5dUtGQUpvQnk2N1N6TGV
2S1d1M3hqYTR5c1RVODRGtkJOY1o3WFNWLXhWQnZGX0NaakhQSmJqbDlCRFVYX1FvOFo2U3pEZEtJZDJFZ2
FTPm13OXNqbjQ3ZVAtVXMyVmpVVkxIOWhVdUZMz29oU0ZOWXdpb01MRW1jQWNINudXTVd4SmpXdy5NZ3VhR
VJJY0NNdE1TM2I2b3N4Y2RR&response_type=code&client_id=8b6b8486-e468-4637-bb50-
b27181d387e4&redirect_uri=https://api.sandboxaccounts.smartfintech.eu/gateway/core/
rest/api/confirmation&scope=AISP&consentId=80374e68-ac03-47b5-bae8-1117b56fc0c9"
        }
    }
}
}

```

2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application





Raiffeisen Online (XS2A via TPP)

Consent Details

TPP Application Name:

SmartAccounts DEV

Access:

Accounts (list of accounts):

- RO03RZBR0000069999999999

Balances (list of accounts with balances):

- RO03RZBR0000069999999999

Transactions (list of accounts with transaction history):

- RO03RZBR0000069999999999

Recurring/One-time consent:

Recurring

Valid until:

2023-05-08T23:59:00Z

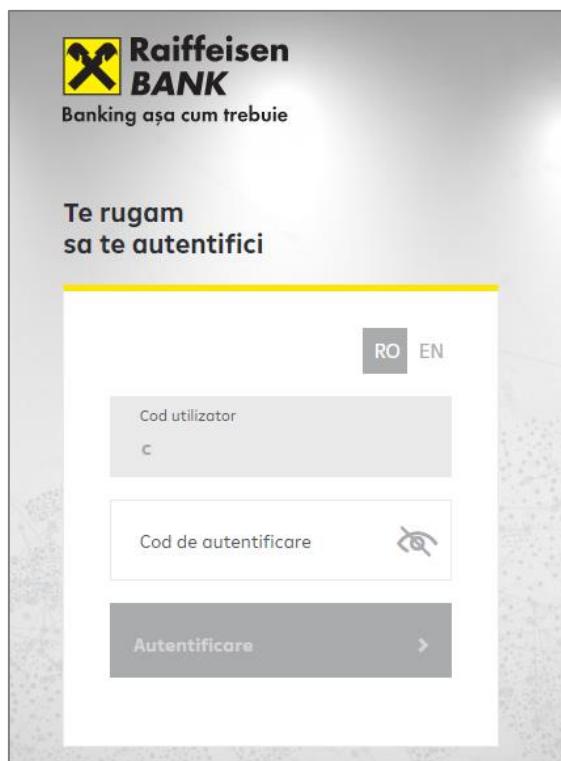
Frequency per day:

4

Approve

After pressing "Approve", thus authorising the consent for all available accounts, developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application



After completing fields "Cod utilizator" and "Cod de autentificare" and pressing "Autentificare", thus authorising the consent for all available accounts, PSU will be redirected to the established page.

3. GET Consent

The screenshot shows the Postman interface with the following details:

- Request Method:** GET
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/consent/RZB/{{internalConsentId}}
- Headers Tab:** Selected. Contains the following entries:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
PSU-IP-Address	192.12.123.32		
Key	Value	Description	
- Other Tabs:** Params, Authorization, Body, Pre-request Script, Tests, Settings, Cookies

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO03RZBR0000069999999999",
          "currency": "RON"
        }
      ],
      "balances": [
        {
          "iban": "RO03RZBR0000069999999999",
          "currency": "RON"
        }
      ],
      "transactions": [
        {
          "iban": "RO03RZBR0000069999999999",
          "currency": "RON"
        }
      ],
      "availableAccounts": null
    },
    "consentStatus": "valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-08",
    "validUntil": "2023-05-08",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

4. GET Consent Status

The screenshot shows a Postman request configuration for a GET operation. The URL is `({{API_BASE_URI}})/gateway/core/rest/api/consent/RZB/{internalConsentId}/status`. The Headers tab is selected, showing the following entries:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32	
Key	Value	Description

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "valid"
  }
}
```

5. GET Accounts

The screenshot shows a Postman request configuration for a GET operation. The URL is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/RZB/{internalConsentId}`. The Headers tab is selected, showing the following entries:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32	
Key	Value	Description

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "add9bbec99be99999eb999999d99999",
      "iban": "RO03RZBR0000069999999999",
      "currency": "RON",
      "name": "My current account",
      "product": "Current Account",
      "cashAccountType": "CACC",
      "status": "enabled",
      "bic": null,
      "accountOfficerDescription": null,
      "openingDate": null,
      "customerSector": null,
    }
  ]
}
```

```

    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "RON",
                "amount": 15000.36
            },
            "balanceType": "expected",
            "referenceDate": "2018-04-02",
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
}
]
}
}

```

6. GET Balances

The screenshot shows a Postman interface with the following details:

- Method:** GET
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/RZB/9999f999a999999d9999da99e999d999/{{internalConsentId}}/balances
- Params:** Authorization (green dot) is selected.
- Headers:** (8 items listed)
- Body:** None
- Tests:** None
- Settings:** None
- Cookies:** None
- Query Params:**

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description	...	

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO03RZBR0000069999999999",
            "bban": "9999999999",
            "currency": "RON"
        },
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 15000.36
                },
                "balanceType": "expected",
                "referenceDate": "2018-04-02",
                "creditLimitIncluded": null,
                "creditDebitIndicator": null
            }
        ]
    }
}

```

```

        "creditDebitIndicator": null,
        "lastChangeDateTime": null,
        "lastCommittedTransaction": null,
        "transactionStatus": null
    }
]
}
}

```

7. GET Transactions

The screenshot shows the Postman interface with a GET request for the endpoint `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 5. Intermediary API's R... / 9. GetTransactio...`. The 'Params' tab is selected, showing the following query parameters:

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> bookingStatus	booked			
<input checked="" type="checkbox"/> dateFrom	2022-11-10			
<input checked="" type="checkbox"/> dateTo	2023-02-08			
Key	Value	Description		

The screenshot shows the Postman interface with a GET request for the endpoint `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 5. Intermediary API's RZB / 8. GetTransactions`. The 'Headers' tab is selected, showing the following headers:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> next	<code>{{next_page}}</code>				
Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {},
        "transactions": {
            "booked": [
                {
                    "transactionId": "NotProvided",
                    "entryReference": null,
                    "endToEndId": "",
                    "bookingDate": "2023-02-01",
                    "valueDate": null,
                    "executionDateTime": null,
                    "transactionAmount": {
                        "amount": null,
                        "currency": null
                    }
                }
            ]
        }
    }
}
```

```

        "currency": "RON",
        "amount": 120.34
    },
    "transactionType": null,
    "exchangeRate": [],
    "creditorName": null,
    "creditorAccount": {
        "iban": null
    },
    "debtorName": "Test Transaction",
    "debtorAccount": {
        "iban": " RO46RZBR0000069899999985"
    },
    "remittanceInformationUnstructured": "CLARITY",
    "proprietaryBankTransactionCode": null,
    "additionalInformation": null,
    "transactionStatus": null,
    "creditDebitIndicator": null
},
{
    "transactionId": "NotProvided",
    "entryReference": null,
    "endToEndId": "",
    "bookingDate": "2023-01-05",
    "valueDate": null,
    "executionDateTime": null,
    "transactionAmount": {
        "currency": "RON",
        "amount": -10001.0
    },
    "transactionType": null,
    "exchangeRate": [],
    "creditorName": null,
    "creditorAccount": {
        "iban": null
    },
    "debtorName": null,
    "debtorAccount": {
        "iban": null
    },
    "remittanceInformationUnstructured": "r002",
    "proprietaryBankTransactionCode": null,
    "additionalInformation": null,
    "transactionStatus": null,
    "creditDebitIndicator": null
}
],
{
    "pending": [],
    "links": null
},
{
    "balances": []
}

```

```

        }
    }
}
```

8. GET Accounts Details

The screenshot shows the Postman interface with a request configuration for 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 5. Intermediary API's RZB / 10. GetAccountDetails'. The method is 'GET' and the URL is `((API_BASE_URI))/gateway/core/rest/api/accounts/RZB/9999f999a99999d999da99e999d999/({internalConsentId})/details`. The 'Headers' tab is active, displaying the following headers:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "resourceId": "add9bbec99be99999eb9999999d99999",
    "iban": "RO03RZBR0000069999999999",
    "currency": "RON",
    "name": "My current account",
    "product": "Current Account",
    "cashAccountType": "CACC",
    "status": "enabled",
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 15000.36
        },
        "balanceType": "expected",
        "referenceDate": "2018-04-02",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null
      }
    ],
    "sortCodeAccountNumber": null
  }
}
```

9. Delete Consent

The screenshot shows a Postman request configuration for deleting a consent. The method is set to **DELETE**, and the URL is `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/RZB/{{internalConsentId}}`. The **Headers** tab is selected, containing the following data:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32	
Key	Value	Description

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "message": "Consent has been successfully revoked"  
    }  
}
```

10. Refresh Token

The screenshot shows a Postman request configuration for refreshing a token. The method is set to **POST**, and the URL is `((MTLS_BASE_URL))/gateway/authenticate/rest/api/refreshToken`. The **Headers** tab is selected, containing the following data:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> internalConsentId	{{internalConsentId}}	
<input checked="" type="checkbox"/> access_token	{{access_token}}	
Key	Value	Description

The screenshot shows a Postman request configuration for refreshing a token. The method is set to **POST**, and the URL is `((MTLS_BASE_URL))/gateway/authenticate/rest/api/refreshToken`. The **Body** tab is selected, showing the following settings:

- Body Type: form-data
- Content-Type: application/x-www-form-urlencoded
- Raw
- Binary
- GraphQL

The body contains the following data:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> refresh_token	{{refresh_token}}	
<input checked="" type="checkbox"/> client_id	{{client_id-partner}}	
Key	Value	Description

Response:

```
{  
    "status": 200,  
    "messageStatus": "SUCCESS",  
    "payload": {
```

```

    "internalConsentId": 101675,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC14alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTY3ODIsImhd
CI6MTY1NTI5NjQ4MiwianRpIjoiZDRlNmFhMzMmIzMi00OTNmLTk5MTMtZjk3YzFkNjRjZTdlIiwiaXNz
IjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOiI2NmE2ZTiYMC11OWE0LTQ2M2ItYWfmZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNWRkYzAyOWQtM
TU5ZC00OWY1LTg3OGQtYzNjQ0IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjD
LCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsib2ZmbGluzV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VyY2VfYWNjZXNzIjp7InRlc3RjbGllbnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbi
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCisIm1hbmcFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydGFjY291bnRzX2NvbnN1bnRfaWRfY2xhaW06MTAx
Njc1IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjc1IiwiY2xpZW50SWQiOiJ0ZXN
0Y2xpZW50djEiLCJjbGllbnRIB3N0IjoiMTAuMC4yNDguMzIifQ.PjfhsaFOlh1r3hhaTz_N1QoYeYzp3OU9q5Hewdi
j2WID87_WTcn31vTxoH0jGplqJRaMqd9KlqyAn3D9BSnF9_EkV6mMwVedcxX3dZewn1i598zf6-OFWLrceM-
71Cbz7qXvrPeLTZ3v-
Ti8myh_XRiA4bfjaBmTywE51jJaXQjdSEN5ZKBm2x1IrFvNKTz9JvtF1nhCqwQm88GV4WoptKxdYDQMBU69
50USjo7pYYaz7Mvqdgc1O901vDXMly1Ze2jWIGXMKRS1PlvtB5euftY9ji1WEIUxq1GidNZPURDH1g79zF-
9McqW3G8NodUb16nH35xinCY749Gv2KGCCQ",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC13OWNhY
2RmMi1kMjk3LTQ0YWIYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzIyMzksImhdCI6MTY1N
TI5NjQ4MiwianRpIjoiNWQ5MDdhOTMtTZlOC00NTdmLWEyNmItMGMyZmMyM2Q4MmM0IiwiiaXNzIjoi
aHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOiI2NmE2ZTiYMC11OWE0LTQ2M2ItYWfmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6IjVkJGMwMj1kLTE1OWQtNDlmNS04NzhkLWMz
YjU4NmM4MzY0NCIsInNjb3B1IjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY3NSJ9.2g
zDn_gKBNOjuG_OzPlfCx0nG4fNvcmZxdo58v1VSFQ"
  }
}

```

In case the "accessToken" expires, the response will have the following structure:

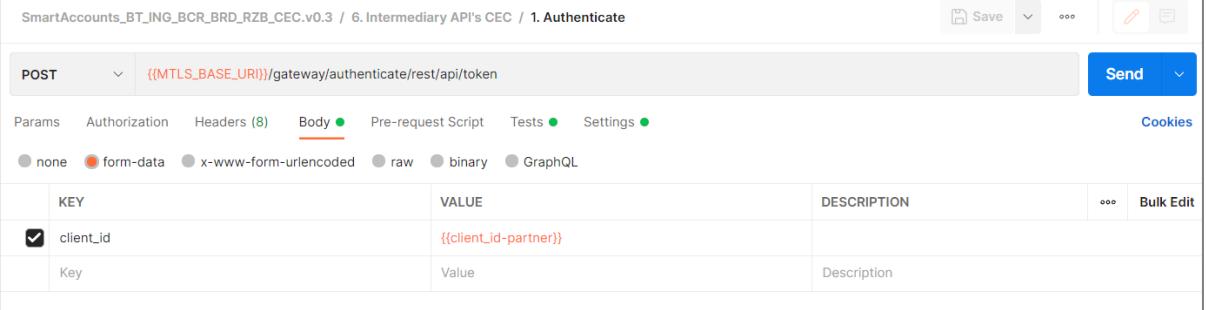
```

{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}

```

7. CEC Bank - Flow

1. Authenticate



The screenshot shows the Postman application interface. The header bar displays "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 1. Authenticate". The main area shows a POST request to `{{MTLS_BASE_URI}}/gateway/authenticate/rest/api/token`. The "Body" tab is selected, showing a key-value pair: "client_id" with value `{{client_id-partner}}`. Other tabs like "Params", "Authorization", "Headers (8)", "Tests", and "Settings" are visible. A "Send" button is at the top right.

Response:

```
{  
    "status": 200,  
    "messageStatus": "SUCCESS",  
    "payload": {  
        "internalConsentId": 101676,  
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS  
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTY4NzksImLhd  
CI6MTY1NTI5NjU3OSwianRpIjoiNDIxOD1kMTAtNzY3ZS00MjY4LWE3NDQtN2FmMzgzzTY0MTNkIiwiAxNz  
IjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI  
sImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC  
J0eXAiOjJCZWfYXZXiIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYjM1YjUxYmMtN  
2RhOS00MzExLWEzXDATN2EwODI0NDAwM2JiIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjD  
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdG1vbiJ  
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2x1cyI6WyJ1bWFfcHJvdGVjdG1vbiJ  
JdfSwiYWNjb3VudCI6eyJyb2x1cyI6WyJtYW5hZ2UtYWNjb3VudCisIm1hbmFnZS1hY2NvdW50LWxpbtzI  
iwidm1ldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnN1bnRfaWRfY2xhaW06MTAx  
Njc2IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjc2IiwiY2xpZW50SWQiOj0ZRN  
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn  
ByZWZlcnJ1ZF91c2VybmFtZSI6InNlcnP2Y2UtYWNjb3VudC10ZXN0Y2xpZW50djEiLCJjbG11bnRBZGRyZ  
XNzIjoiMTAuMC4yNDguMzIifQ.XgNJYqosqS5oC-  
Sun_LwvP_rXFT361zQs08fSnX7ZccYaPpBHYsbP53IvfZdvw4Qv6xatv9d086m-  
0HV2WHiwtDyPekpi8ZFPAPs_jfHR0Brc7NrEkX3c1s6vndlE97-J48eyHyv-  
HadEyl1KZf_jeKJ_8cEzD8aOZiYwy8RXN3gAc0OSm7-2yLxee639IwrOyZ2ir8ok4-  
iLQeElH8owJPPp9jswcrGueuqU9dTxAxNK7oUA_Ue17UDI12-Hw4CAmmvTredhQwaRPKtaQsd9DN1rDXZ-  
gyYirweR5eSjTen6sc28_HAwjpjt42oD0e_C54pVZkzU4D_i8OC9parAJwg",  
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS  
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzI1NzksImLhdCI6MTY1N  
TI5NjU3OSwianRpIjoiMGE4Y2NhYWUtOWJ1OC00ODQ0LWIzZTEtMzlhMja0YjFhZmY4IiwiAxNzIjoiHR0  
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI  
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC  
JzdWIiOii2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY  
XpWIjoiGVzdGNsaWvudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImIzNWI1MWJjLTdkYTktNDMxMS1hMWQwLTdh  
MDgyNDQwMDNiYiIsInNjb3BlIjoiic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY3NiJ9.vg  
m0iIzCYfspshHc27STWnNmMagh-umWPRovgbd_bhbg"  
}  
}
```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / **initConsent**

POST `((API_BASE_URI))/gateway/core/rest/api/initConsent/{{internalConsentId}}`

Params Authorization Headers (11) Body Pre-request Script Tests Settings Cookies

Headers (9 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / **initConsent**

POST `((API_BASE_URI))/gateway/core/rest/api/initConsent/{{internalConsentId}}`

Params Authorization Headers (11) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1
2     "codeBank": "CEC",
3     "periodofValid": 90,
4     "psuEmail": "monica.rotaru@itsmartsystems.eu",
5     "psuIntermediarId": "123",
6     "redirectURL": "https://google.com",
7     "TCaccepted": true
8

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "links": {
      "scaOAuth": {
        "href": "https://api-
test.cec.ro/cec/tpp/oauthcec/oauth2/authorize?state=ZX1KbGJtTWlPaUpCTVRJNFEwSkRMVWh
UTWpVMk1pd21ZV3huSWpvaVpHbH1JaXdpWTNSNULqb21TbGRVSW4wLi5aaVdtWHQ3bmg2Q2J4OTNrd3AxSD
JRLjJDT0Q1VDVqd0ZXX1pJNEpIbUx5MUZLRlNycjViTwdTaWZSRWpWRVRhRGtmOU11NWxaay1uUnk4RWRid
ldoS0RYV2FwcW1LQzVfNTJrZTI0YWVodFdzcWJHWmlJOUNLVuhnRHppeUpBTWMwMkk1WTJ2VU9GaX1ja01E
VlpTb2ExTVUzWW15M3F4MV15eDV0z2RwZHFTTUcxbW9IYjJiWndpYzYwMjBNODhLSjNDb0t3cWdGVVd6cTh
FbWhIdDhIRWx5ZE92dUE5ZnpxF15eTV6OFN4eTh3ej1EWEpDdnVYTHE2eDFOaT1CakJWd2E3ZFZ0UzzvU
lNX2pQS0FhSwdvd0dKY2Jac091SzluRE9nOTdPRjRocDFVcnPgwVFTeHphRzNLQ1ZjUnJ0UFhmczNJcjdyV
VZpUk4ydnpybUkwZkR5M31CcFAyVzJXM1RncVEwb0I4bDdj1Y4dHBobF1LvmJsYzdLTnYtcX16cHZNcX1O
U2ROYS13YWVmDWRaTZ0T01TZ0FRVzJmMF9GQ1drV01UVWk0dndoeTlsMnRqcD10MV9GUEwwY2xqUmIxcoI
yUnJ6OTR0LUg4ckNRMkF5TDQ5bTRfV1c5dVFnZ0pWMW9jZ21qbC1KTUU5TUh1QWIzbHVTZG5UTHVkcHZTQ0
1tWnlmS1dEX2ZZR051STdBQWtHVXZrN1VBYjdxYkcwdjJWWjh2RHNOdDJaalU5VWR3R0ZYTk1QQ0Jmb1NHc
kRFTGYYWUtmoW5QVFJ1eFZHNjBLamE2RUJ1RFhndG1LUEZZNVg4cUtpVDNHQWhjRjBODndkaVR1aG5xdEVE
M1k4dnJ4YVFqcUjkMW9EQTVzUFNVMG10Y1V0RTNLukJEVm1SaWdRMmNCTzJ6YklHaFZlWTkxVTBtOFNkR3Z
0dExraEJGQXB1cVdNVUh5SjRJdElTT1AyenNSZfH4N3h1MHFpRTJkekI3WWJ1TDFrOGhCNEZ2bnhmeU51SE
NzUVJsOVExTFJ1VzlqbX1BaGRPMFVyU1VOR2dyRk51c1E2UHFoUj1SU01DReptOTRTNG1rNGlubTQ2b2JKR
```

```

VFCMV9Gdn1UXzBuMmxpekJLdjB6NU5peEQtb0hDS1JEVnBaaVNINTFQaWxXXzAxYnJWLvhTT2V2bzVLTk1S
Q09jcmNNX3I5aEtXQWdjV05wem5zQ2FORm9PdUliX3UtYzIzRkRBNTI3REhPb1M0alN5Z1Bjam9nSUdCcW4
zQWREUEFsWEtsQ0hNcndHQy1sVnQzQTlRbmRUSlhZTHB1Sm1SbUFiVFg5bEE2Yk4yTDFuNnE1ZE44VV8wdW
dnsjBOVEFvv0MtZEWcEZIOW0ySnhWRHdfWGtfekpGb292aTRaQ1FuLVZKRxp1VnVJZGpuMkNybjBSQVhoS
ERTQ113TzhXSmZFd1GVjdJNnFCU1F5SzHZEl1aR2VNUddMRl9wUVJXcHc4ajB1QXBGaXdFaXVmN0tESzE1
dlgwY2FSWedkSW56Y212cXVsb2swVEZScf82WFJHY1ZTdmFfYXZPQ1Nbc2k0R3hvQvhVQxdXOEVIY1FDZnh
BUkhpVy0zSkloRGpXc0kuNHJxc1VqVGJrTGJrTjBMSkZVSXZ6dw&response_type=code&client_id=55
01752f-a17a-45b1-8fe4-
c241bae0f73f&redirect_uri=https://api.sandboxaccounts.smartfintech.eu/gateway/core/
rest/api/confirmation&scope=AIS:dcafe57e-eca7-11ec-8283-
0a0800270000&code_challenge=7LOaXFoz3012CEnUkWekNcCvmlk5YK6nMg2Fbts&code_challenge_
method=S256"
}
}
}
}

```





Solicitare acord accesare conturi

Bine ati venit, **testcecapi**

Aplicatia **SmartAccounts TEST** SOLICITA permisiunea de a accesa conturile dvs. deschise la CEC Bank cu scopul :

AIS Vizualizare date conturi

Selecteaza / Deselecteaza tot

Conturi

- RO42CECEQW9898bON9993644
- RO49CECER79292bON8883343
- RO12CECER792R8Ueb8883338
- RO03CECER792R7SXV6635199
- RO46CECEQW93R8Ueb9357561

Balante

- RO42CECEQW9898bON9993644
- RO49CECER79292bON8883343
- RO12CECER792R8Ueb8883338
- RO03CECER792R7SXV6635199
- RO46CECEQW93R8Ueb9357561

Tranzactii

- RO42CECEQW9898bON9993644
- RO49CECER79292bON8883343
- RO12CECER792R8Ueb8883338
- RO03CECER792R7SXV6635199
- RO46CECEQW93R8Ueb9357561

Actionati [Permit Acces!] daca sunteți de acord sau [Nu sunt de acord!] daca nu sunteți de acord.

Apasand Permit Acces! veti fi redirectat la adresa

<https://api.sandboxaccounts.smartfintech.eu/gateway/core/rest/api/confirmation>.

Nu sunt de acord!

Permit Acces!

After selecting at least one account and type of consent, and pressing "Permit access!", thus authorising the consent based on granularity (which account, balances and transactions), PSU or developer will be redirected to the established page.

3. GET Consent

The screenshot shows a Postman interface with the following details:

- Request Method: GET
- URL: `((API_BASE_URI))/gateway/core/rest/api/consent/CEC/({internalConsentId})`
- Headers Tab: Selected. Contains:
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Body, Pre-request Script, Tests, Settings, Cookies tabs are visible but not selected.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "access": {  
            "accounts": [],  
            "balances": [],  
            "transactions": [],  
            "availableAccounts": null  
        },  
        "consentStatus": "valid",  
        "frequencyPerDay": null,  
        "lastActionDate": "2023-02-08",  
        "validUntil": "2023-05-09",  
        "recurringIndicator": false,  
        "combinedServiceIndicator": null,  
        "creationDate": null  
    }  
}
```

4. GET Consent Status

The screenshot shows a Postman interface with the following details:

- Request Method: GET
- URL: `((API_BASE_URI))/gateway/core/rest/api/consent/CEC/({internalConsentId})/status`
- Headers Tab: Selected. Contains:
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Body, Pre-request Script, Tests, Settings, Cookies tabs are visible but not selected.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "consentStatus": "valid"  
    }  
}
```

```

        }
    }
}
```

5. GET Accounts

The screenshot shows the SmartAccounts_BI_Ing_BCR_BRD_RZB_CEC.v0.3 API tool interface. A GET request is being prepared to the endpoint `/gateway/core/rest/api/accounts/CEC/{{internalConsentId}}`. The Headers tab is active, displaying the following configuration:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "207940324",
      "iban": "RO26CECER999T4ecT8982492",
      "currency": "USD",
      "name": "410.21147840.207940324.USD",
      "product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - USD",
      "cashAccountType": "CurrentAccount",
      "status": null,
      "bic": null,
      "accountOfficerDescription": null,
      "openingDate": null,
      "customerSector": null,
      "customerType": null,
      "accountUsage": null,
      "workingBalance": null,
      "onlineActualBalance": null,
      "balances": [
        {
          "balanceAmount": {
            "currency": "USD",
            "amount": 0.0
          },
          "balanceType": "closingBooked",
          "referenceDate": null,
          "creditLimitIncluded": null,
          "creditDebitIndicator": null
        }
      ],
      "sortCodeAccountNumber": null
    },
    {
      "id": "207940324"
    }
  ]
}
```

```

    "resourceId": "25511978",
    "iban": "RO42CECEQW9898bON9993644",
    "currency": "RON",
    "name": "421642",
    "product": "Cont curent - persoane fizice - LEI",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "RON",
                "amount": -32289.31
            },
            "balanceType": "closingBooked",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
},
{
    "resourceId": "207940308",
    "iban": "RO83CECER999b2XeV8982495",
    "currency": "HUF",
    "name": "410.21147840.207940308.HUF",
    "product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - HUF",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "HUF",
                "amount": 0.0
            },
        }
    ]
}

```

```

        "balanceType": "closingBooked",
        "referenceDate": null,
        "creditLimitIncluded": null,
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
},
{
    "resourceId": "100249515",
    "iban": "RO49CECER79292bON8883343",
    "currency": "RON",
    "name": "411.469952.2227767.RON",
    "product": "Cont curent - persoane fizice - LEI",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "RON",
                "amount": 0.0
            },
            "balanceType": "closingBooked",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
},
{
    "resourceId": "100249740",
    "iban": "RO12CECER792R8Ueb8883338",
    "currency": "EUR",
    "name": "411.469952.2227772.EUR",
    "product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - EURO"

    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,

```

```

    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "EUR",
                "amount": -41.15
            },
            "balanceType": "closingBooked",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
},
{
    "resourceId": "201415500",
    "iban": "RO03CECER792R7SXV6635199",
    "currency": "CHF",
    "name": "421.469952.4475900.CHF",
    "product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - CHF",
    "cashAccountType": "CurrentAccount",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "CHF",
                "amount": 0.0
            },
            "balanceType": "closingBooked",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
},
{
    "resourceId": "33909743",
    "iban": "RO46CECEQW93R8Ueb9357561",

```

```

"currency": "EUR",
"name": "031.88924.753549.EUR",
"product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - EURO"

,
"cashAccountType": "CurrentAccount",
"status": null,
"bic": null,
"accountOfficerDescription": null,
"openingDate": null,
"customerSector": null,
"customerType": null,
"accountUsage": null,
"workingBalance": null,
"onlineActualBalance": null,
"balances": [
{
    "balanceAmount": {
        "currency": "EUR",
        "amount": 0.0
    },
    "balanceType": "closingBooked",
    "referenceDate": null,
    "creditLimitIncluded": null,
    "creditDebitIndicator": null
}
],
"sortCodeAccountNumber": null
},
{
"resourceId": "207940306",
"iban": "RO33CECER999R9WRP8982497",
"currency": "GBP",
"name": "410.21147840.207940306.GBP",
"product": "Cont curent - pers. fizice REZIDENTE si NEREZIDENTE - GBP",
"cashAccountType": "CurrentAccount",
"status": null,
"bic": null,
"accountOfficerDescription": null,
"openingDate": null,
"customerSector": null,
"customerType": null,
"accountUsage": null,
"workingBalance": null,
"onlineActualBalance": null,
"balances": [
{
    "balanceAmount": {
        "currency": "GBP",
        "amount": 0.0
    },
    "balanceType": "closingBooked",
}
]
}

```

```

        "referenceDate": null,
        "creditLimitIncluded": null,
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
},
{
"resourceId": "2641063123453605",
"iban": "RO87CECES999164899997299",
"currency": "RON",
"name": "4243XXXXXXXXX1287",
"product": "VISA Debit - Staff",
"cashAccountType": "CurrentAccount",
"status": null,
"bic": null,
"accountOfficerDescription": null,
"openingDate": null,
"customerSector": null,
"customerType": null,
"accountUsage": null,
"workingBalance": null,
"onlineActualBalance": null,
"balances": [
{
"balanceAmount": {
"currency": "RON",
"amount": 1242.01
},
"balanceType": "closingBooked",
"referenceDate": null,
"creditLimitIncluded": null,
"creditDebitIndicator": null
}
],
"sortCodeAccountNumber": null
}
]
}

```

6. GET Balances

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 7. GetBalances'. The request is a GET method to the URL `{(API_BASE_URI)}/gateway/core/rest/api/accounts/CEC/25511978/{internalConsentId}/balances`. The 'Params' tab is selected, showing a single query parameter 'Key' with the value 'Value'. Other tabs include 'Authorization', 'Headers (8)', 'Body', 'Pre-request Script', 'Tests', 'Settings', and 'Cookies'.

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description	...	

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO42CECEQW9898bON9993644",
            "currency": "RON"
        },
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": -32289.31
                },
                "balanceType": "closingBooked",
                "referenceDate": "2023-02-08",
                "creditLimitIncluded": null,
                "creditDebitIndicator": null,
                "lastChangeDateTime": null,
                "lastCommittedTransaction": null,
                "transactionStatus": null
            }
        ]
    }
}
```

7. GET Transactions

The screenshot shows the Postman interface with the following details:

- Request URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/CEC/25511978/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2022-04-10&dateTo=2022-05-12
- Method:** GET
- Params:** bookingStatus (checked), dateFrom (checked), dateTo (checked)
- Headers:** (10 headers listed but not detailed in the screenshot)
- Body:** (Empty)
- Tests:** (Empty)
- Settings:** (Empty)
- Cookies:** (Empty)

The screenshot shows the Postman interface with the following details:

- Request URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/CEC/25511978/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2022-03-10
- Method:** GET
- Params:** (Empty)
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - 9 hidden headers
- Body:** (Empty)
- Tests:** (Empty)
- Settings:** (Empty)
- Cookies:** (Empty)

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO42CECEQW9898bON9993644"  
        },  
        "transactions": {  
            "booked": [  
                {  
                    "transactionId": "1250649778",  
                    "entryReference": null,  
                    "endToEndId": null,  
                    "bookingDate": "2023-02-07",  
                    "valueDate": "2023-02-07",  
                    "executionDateTime": null,  
                    "transactionAmount": {  
                        "currency": "RON",  
                        "amount": 39.35  
                    },  
                    "transactionType": null,  
                    "exchangeRate": [],  
                    "creditorName": null,  
                    "creditorAccount": {  
                        "iban": null  
                    },  
                    "debtorName": null,  
                    "debtorAccount": {  
                        "iban": null  
                    },  
                    "remittanceInformationUnstructured": "Tupunuru vyrgmunt - qsopu  
ryru solt turytor",  
                    "proprietaryBankTransactionCode": null,  
                    "additionalInformation": null,  
                    "transactionStatus": null,  
                    "creditDebitIndicator": null  
                },  
                {  
                    "transactionId": "896970683",  
                    "entryReference": null,  
                    "endToEndId": null,  
                    "bookingDate": null,  
                    "valueDate": "2022-09-27",  
                    "executionDateTime": null,  
                    "transactionAmount": {  
                        "currency": "RON",  
                        "amount": 130.5  
                    },  
                    "transactionType": null,  
                    "exchangeRate": []  
                }  
            ]  
        }  
    }  
}
```

```

        "creditorName": null,
        "creditorAccount": {
            "iban": null
        },
        "debtorName": "",
        "debtorAccount": {
            "iban": ""
        },
        "remittanceInformationUnstructured": "teste dbron",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
],
"pending": [],
"links": null
},
"balances": []
}
}

```

8. GET Transaction Details

The screenshot shows a REST API client interface. The URL is `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 9. GetTransactionDetails`. The method is set to `GET`. The Headers tab is active, showing the following configuration:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "transactionId": "1227476280",
        "entryReference": null,
        "endToEndId": null,
        "bookingDate": "2022-03-10",
        "valueDate": "2022-03-10",
        "transactionAmount": {
            "currency": "RON",
            "amount": -18.55
        },
        "transactionType": null,
        "creditorName": null,
        "creditorAccount": {

```

```

        "iban": null
    },
    "debtorName": null,
    "debtorAccount": {
        "iban": null
    },
    "remittanceInformationUnstructured": "Sqpptyqlyzqru torqntq-
Qsxytqru torqntq nutq lq ssqtuntq tyn sont surunt puntru sontrqstul nr 894414336",
    "executionDateTime": null
}
}

```

9. GET Accounts Details

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 10. GetAccountDetails

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/CEC/25511978/{{internalConsentId}}/details

Headers (8)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "resourceId": "25511978",
        "iban": "RO42CECEQW9898bON9993644",
        "currency": "RON",
        "name": "421642",
        "product": "Cont curent - persoane fizice - LEI",
        "cashAccountType": "CurrentAccount",
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": -32289.31
                },
                "balanceType": "closingBooked",
            }
        ]
    }
}
```

```

        "referenceDate": "2023-02-08",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
}
}

```

10. Refresh Token

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Headers (10) Body Pre-request Script Tests Settings Cookies

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> internalConsentId	{{{internalConsentId}}}		...		
<input checked="" type="checkbox"/> access_token	{{{access_token}}}		...		
Key	Value	Description	...		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC.v0.3 / 6. Intermediary API's CEC / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	{{{refresh_token}}}		...	
<input checked="" type="checkbox"/> client_id	{{{client_id-partner}}}		...	
Key	Value	Description	...	

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 101676,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NTUyOTcyNjIsImhdCI6MTY1NTI5Njk2MiwianRpIjoiN2NlNDE0OGUtNWI3Ni00N2M4LWF1YWMtNDNiMzNkNWN1Y2VjIiwiAxNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYjM1YjUxYmMtN2RhOS00MzExLWEzZDAwN2EwODI0NDAwM2JiIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VyY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcyI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcyI6WyJtYW5hZ2UtYWNjb3VudCIisIm1hbmcFnZS1hY2NvdW50LWxpbtzI
```

```

    iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTAx
    Njc2Iiwic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTAxNjc2IiwiY2xpZW50SWQiOjZ0ZXN
    0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4yNDguMzIiLCJ1bWFpbF92ZXJpZml1ZCI6ZmFsc2UsIn
    ByZWZ1cnJ1ZF91c2VybmFtZSI6InNlcnPZy2UtYWNjb3VudC10ZXN0Y2xpZW50djeiLCJjbG11bnRBZGRyZ
    XNzIjoiMTAuMC4yNDguMzIifQ.CzVeGh18cZ1hcTRyZTwm61W2AZ178x9XgqM4QWtUEt1QDwytzCKpTFNAL
    w1DXYB1_ZF6sbK6FoshINyoK0qtAXEVd3OFapV45RonOJfSiYyi-
    41daTXP31s0Qk2HSHkLDerPTaqybKmBn4xyUyPiCZX8ImgoinW4QqtY4DAcrWVldH0g75hzdnxEIP6X2hzd
    pB3Y1Ig5bcaSzWDr4rPNy5v_Ms5kVMFHix30JOpg4xxK11ZGWvvPG_2SQHFew3s3tB00_1-
    PZSxxBc7RmMsHzcFRZHiMumEhha2GCVAr635OPot9Pleogzjsm0nREtYNp0mSljFCj7mTVSPpqKwgOg",
        "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwa2lkIia6IC13OWNhY
    2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2NjMwNzI1NzksImhdCI6MTY1N
    TI5Njk2MiwianRpIjoiNDRmMDE4M2Ut0GR1Yy00MGQyLWE1ZDITMTEzTY4Njc2NzI4IiwiiaXNzIjoiAHR0
    cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oJgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
    6Imh0dHA6Ly9rZX1jbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
    JzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFmZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
    XpWIjoidGVzdGNsaWvudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImIzNWI1MWJjLTdkYTktNDMxMS1hMWQwLTdh
    MDgyNDQwMDNiYiIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjEwMTY3NiJ9.Mq
    Q7TmT93oVCoeR3A9m-zeDxaqUjro96TUnHbeBli1U"
    }
}

```

In case the "accessToken" expires, the response will have the following structure:

```
{
    "status": 401,
    "messageStatus": "Unauthorized",
    "moreDetails": {
        "text": "Invalid access token"
    }
}
```

8. First Bank - Flow

1. Authenticate

The screenshot shows a Postman interface with the following details:

- Method:** POST
- URL:** {{MTLS_BASE_URL}}/gateway/authenticate/rest/api/token
- Body:** form-data (selected)
- Params:** none
- Headers:** (8) (selected)
- Body Content:**

KEY	VALUE	DESCRIPTION	...	Bulk Edit
client_id	{{{client_id-partner}}}			
Key	Value	Description		
- Send** button

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116298,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZZJn0.eyJleHAiOjE2NzU4NTk2MTksImhd
CI6MTY3NTg1OTMxOSwianRpIjoiNmE2ZTQ0MDQtNTEExMi00MzQ1LWJhMTgtZjB1MjkyMGFkMTgzIiwi
aXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cy
I
sImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfYzxIiLCJhenAiOj0ZN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYTZmMGUxMDktOT
TBkZS00MmE5LWE3YzAtZDI0MzM4MWE0ZTE2IiwiYWNyIjoiMSI
sImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZv9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCISIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUiOjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTE2
Mjk4IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYwltIjoiMTE2Mjk4IiwiY2xpZW50SWQiOj0ZN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xNzYuMTg4IiwiZw1haWxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOjZzXJ2aWN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.DTZcfQkCcSAXUcp3qViueUjeyELaE1XqbD79U9q99VbViVmSuSEKwO-
FZh866qvB9AHnNcJ3ukSHxGYy9omEEuY07PLRjfBJz9hC4JESNbhfEaLQbRXVGrt61Z5pXXHn6qVZXSy90
5I0r0mFMZ-b3cGM_GzA21FM9_AUPWzB7FNBMjd6DegN9CTdR5rQ81ZDKX_AzOXbXiagIrJOj-v-
NQTfqB4TUrchAlyahF11KNMzvfW1zj1Tul8oDDwEuxMF9cZi83pmx1iT-
sBZFA91s5ocxLQs1PjU7L6bH7NsDPeLgpi2rxHvpZdE-wxfEQ2v_e1ViH0teIIEMSqIB-Q",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhY
2RmMi1kMjk3LTQ0YWIYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM2MzUzMTksImhdCI6MTY3N
Tg1OTMxOSwianRpIjoiMzM2ZTg3ZWUtOTJkZS00NjUxLWFmNGEtZjJkNmE3YjE3ZDNjIiwi
aXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cy
sImF1ZCI6Imh0dHA6Ly9rZX1jbG9haylhCAtc2VydmljZTo4MDgwL2F1dGgvcvMvhbG1zL3NtYXJ0YWNjb3VudH
MiLCJzdwIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51Iiwi
YXpwljoiGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImE2ZjB1MTA5LTkwZGutNDJhOS1hN2MwLWQy
NDMzODFhNGUxNiIsInNjb3BlIjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYwltOjExNjI5OCJ9.Qf
Fsre601_FZOHxvy3ofphFYkqF7DAJ57k-3pCgomfw"
  }
}

```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 4. InitConsent

POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/({{internalConsentId}})

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

Headers (8 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 4. InitConsent

POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/({{internalConsentId}})

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

Body (raw, JSON, GraphQL, Binary, Text, CSV, XML, YAML, Beautify)

```

1
2     "codeBank": "FB",
3     "periodOfValid": 90,
4     "psuEmail": "aaa.aaaa@mail.com",
5     "psuIntermediarId": "123",
6     "redirectURL": "https://google.com",
7     "TCaccepted": true
8

```

Response:

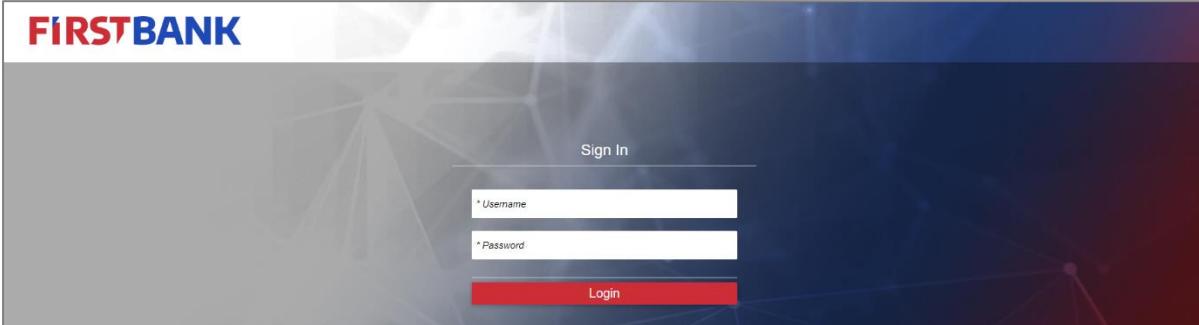
```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://dev.firstbank.ro/openbanking/services/startAuthori
ze?state=eyJlbmMiOiJBMTI4Q0JDLUhTMjU2IiwiYXnIjoiZGlyIiwiY3R5IjoisldUIIn0..MItgkVop-
kLwbY8pw9gQRA.f0vZg0ka5h-
a5bTFQOS3o8r3AAFBaaEu4TmwVQlPPP628_SST09FBzSi3DpATTSQxivZixRdxBPln0fPsQO6rxylKPTbwx
ZqsLTYXkCyX61Ei8fl2GNs3m37r_IgnaWeRZ_8cBRTBdbsQ1rAOts4WNhdC2ExQnQoR8iixfxUtfAjKFS5P
cill7dALV3UtFviuPvZAP2NZTs-
AHaD58qiOCiILjP6OU1B8u_2yTvz5D_sJFDVFvujRdDqUa6KK13j_vutPwYinAq4HOV-RwqiDBounWxm-
ZHgzP5GcaWIsvJVWIV6k3Q32UP_xTGid8G63RaFnGJvkkBk7DHTkWbSvBtQoWJLQapwKK4Z-
kEbHuAlNLJbWvdizOuqCumev10GhVGEiG8nPxDz6K6npqKPiU0gLKPZalPSuKgPff4FauPqjreLkOBsAW-
AzbWGNQUn1XF8m4vPhs8-dqe4Oj20kV7QNx29c7nS-zPAHnhYbheOe-
NBDBNnmnUiYrcPvy0NXUurMFVYd1ROz3W--
Tbwf3dIcQcp_43Fd3KEaCmuW_031ouTKF5UV8ysEpoCL9UF94s1kNN13_OEbaRBfdf035T3_8t3eK81Ff-
f4oKCAltFb1_v8OP4KYDIe_-
i4K8H2VSVpLs_oPIOdkW91xGoo_m5bAvY94rhz7MN384heXdVXIUt08VO6-vrzRpuK6sZyGiks-
hOpk4Dc15FC642HbG4mVORx7I9v0XdAB0hHaoJY1DjdYgcWF116bywPrBT6tc4y9eNDEpgN-
70No05IMK5S4xyhtox9fEA2x432CS9WrjLojWvxm2ZhS9Eg1JZXn18PG7Cw2kZDmdbSI-
exY2uqkOHTUk9ZDHHLRhgiezlOjXXuOXpMLZ3tdZwAk_V-f1ZMLioww-EGXp2Pbyh3dxRKU9UL-
GSxlEysCp8T3o0qsorXvD9px4GR4UXmUaavqMy0hEbOweXG-A3YWqmZ00wAhAtWmD-
```

```

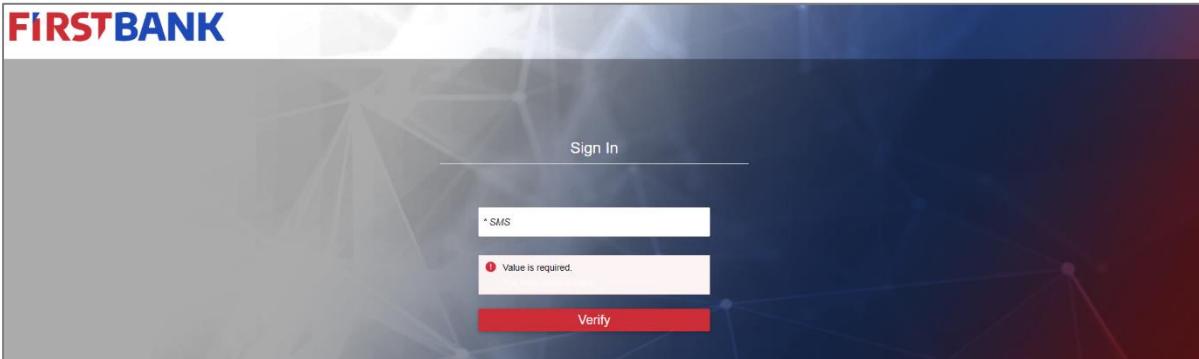
rXxL0Z6aVzHzOuY2yteg6Cjf2cLV6oZnY2gSrTZkzTVKD1v7TmDgWkzwxjgRk24yuqqPwNHJEsnHdqL2HBE
7xb-
HfiPAbAgT.rXDPDwzja4AU5LaCcs_Stg&response_type=code&client_id=toMRlUX5q2anHb2PYevLI
CNB7xxmUFjv&scope=transactions_balances_accounts&consent_id=0e7a00c4-cdd3-4bc9-
af13-5abbd1104e42"
}
}
}
}

```

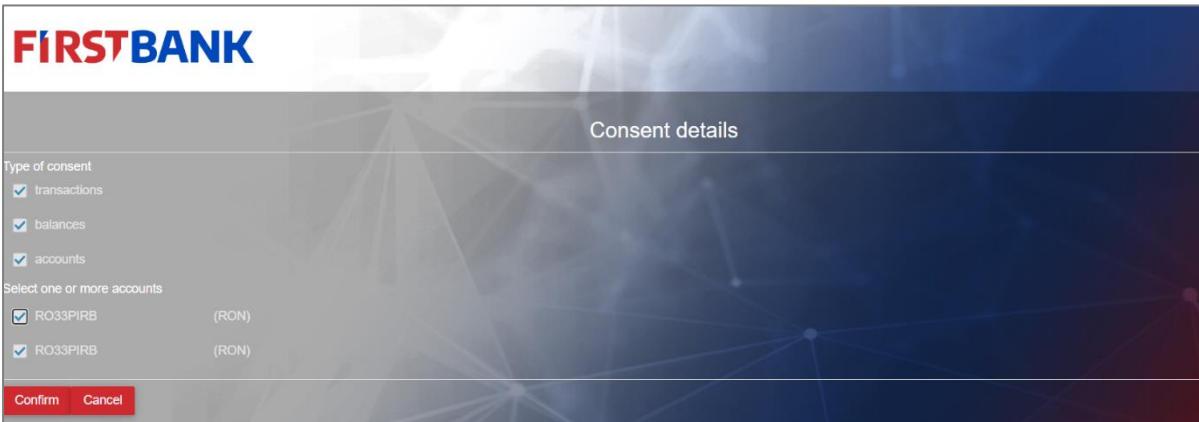
2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application



The screenshot shows the FIRSTBANK sign-in page. The header reads "FIRSTBANK". Below it is a "Sign In" form with two input fields: "Username" and "Password", both marked with an asterisk (*) indicating they are required. A red "Login" button is at the bottom of the form.



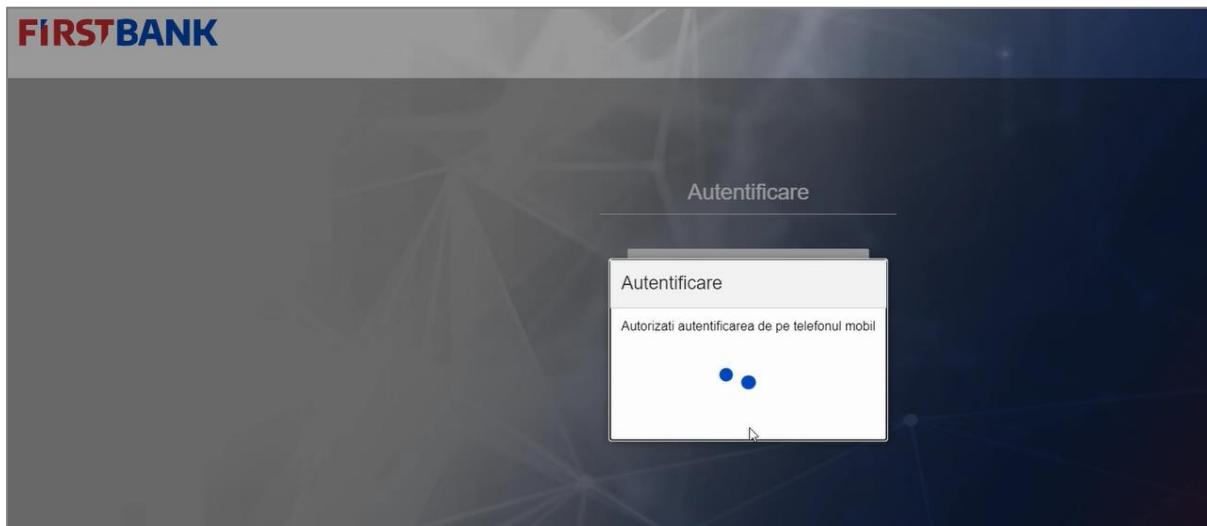
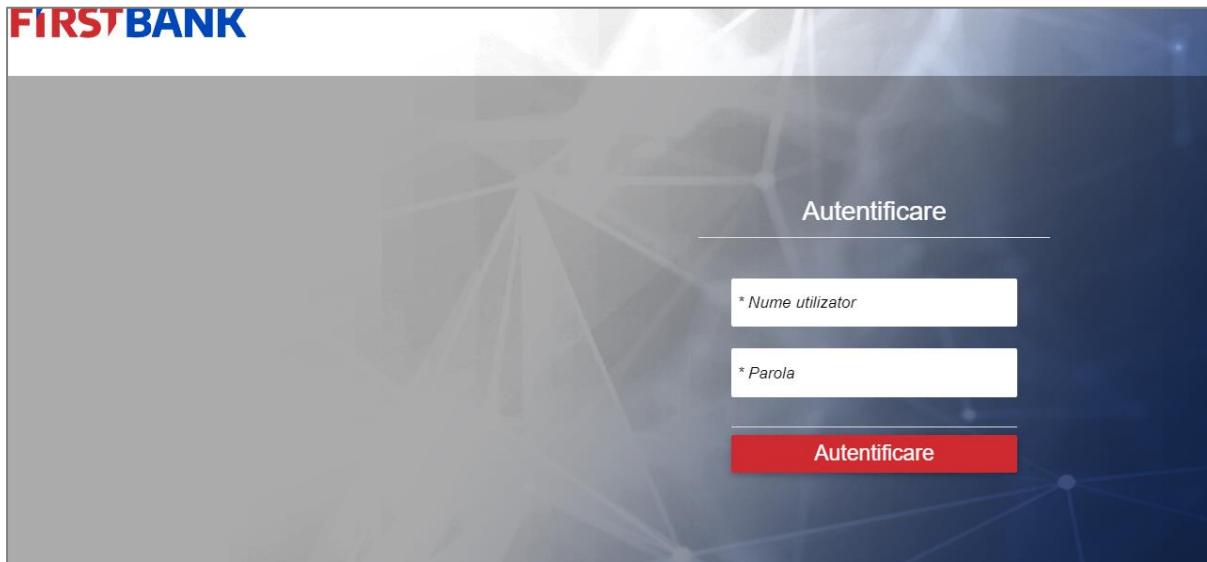
This screenshot shows the same sign-in page as above, but with an error message. The "SMS" field is highlighted in red, and a tooltip says "Value is required." A red "Verify" button is at the bottom.

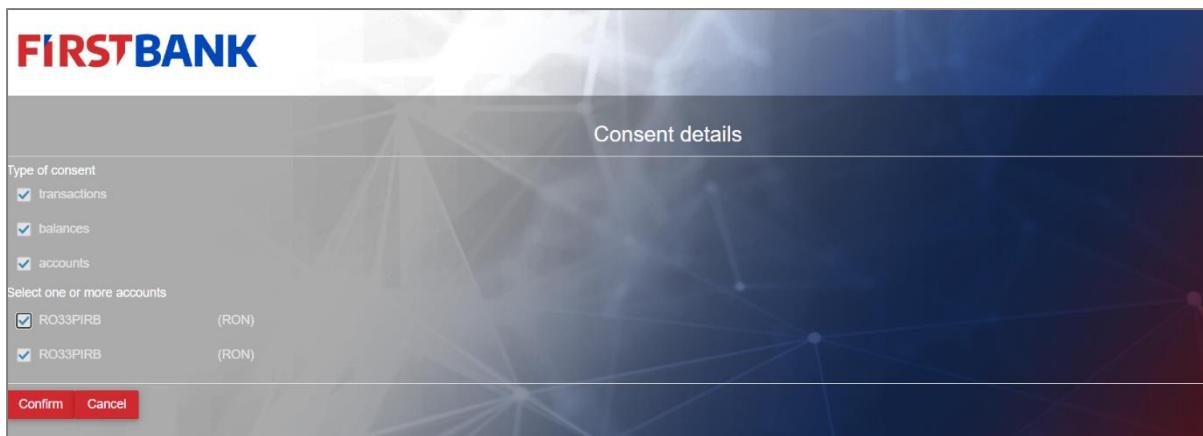


The screenshot shows the "Consent details" page. It includes sections for "Type of consent" (with checkboxes for "transactions", "balances", and "accounts", all of which are checked), "Select one or more accounts" (with checkboxes for "RO33PIRB" and "RO33PIRB" (RON), both checked), and a "Confirm" and "Cancel" button at the bottom.

After selecting at least one account and type of consent, and pressing "Confirm", thus authorising the consent based on granularity (which account, balances and transactions), developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application





After selecting at least one account and type of consent, and pressing "Confirm", thus authorising the consent based on granularity (which account, balances and transactions), PSU will be redirected to the established page.

3. GetConsent

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO33PIRB42191A4819001000"
        },
        {
          "iban": "RO33PIRB42191A4819001001"
        }
      ],
      "balances": [
        {
          "iban": "RO33PIRB42191A4819001000"
        },
        {
          "iban": "RO33PIRB42191A4819001001"
        }
      ],
      "transactions": [
        {
          "iban": "RO33PIRB42191A4819001000"
        },
        {
          "iban": "RO33PIRB42191A4819001001"
        }
      ]
    }
  }
}
```

```

        "transactions": [
            {
                "iban": "RO33PIRB42191A4819001000"
            },
            {
                "iban": "RO33PIRB42191A4819001001"
            }
        ],
        "availableAccounts": null
    },
    "consentStatus": "valid",
    "frequencyPerDay": 4,
    "lastActionDate": "2023-02-08",
    "validUntil": "2023-05-08",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
}
}

```

4. GetConsentStatus

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "valid"
    }
}

```

5. GetAccounts

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": [  
        {  
            "resourceId": "276c8330-37ed-4b76-9789-c0278c4570d0",  
            "iban": "RO33PIRB42191A4819001000",  
            "currency": "RON",  
            "name": "Popescu Ion",  
            "product": null,  
            "cashAccountType": "*",  
            "status": null,  
            "bic": null,  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        },  
        {  
            "resourceId": "aca299c1-8a75-4fd7-b280-1d663bfbfc3e",  
            "iban": "RO33PIRB42191A4819001001",  
            "currency": "RON",  
            "name": "Popescu Ion",  
            "product": null,  
            "cashAccountType": "*",  
            "status": null,  
            "bic": null,  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        }  
    ]  
}
```

6. GetBalances

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 8. GetBalances

GET https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/FB/a1b1db08-b881-44d9-b5ce-9c8880c71314/{{internalConsentId}}/balances

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO33PIRB42191A4819001000"
    },
    "balances": [
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 163079.18
        },
        "balanceType": "closingBooked",
        "referenceDate": "2018-05-11",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null,
        "lastChangeDateTime": null,
        "lastCommittedTransaction": null,
        "transactionStatus": null
      }
    ]
  }
}
```

7. GetTransactions

GET https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/FB/a1b1db08-b881-44d9-b5ce-9c8880c71314/{{internalConsentId}}/transactions?bookingStatus

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> next	{{next_page}}				
Key	Value	Description			

Postman screenshot showing a GET request configuration. The URL is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/FB/a1b1db08-b881-44d9-b5ce-9c8880c71314/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-01-01&dateTo=2023-02-08`. The 'Params' tab is selected, displaying three query parameters: bookingStatus (checked, value: booked), dateFrom (checked, value: 2023-01-01), and dateTo (checked, value: 2023-02-08). A 'Query Params' table is also present.

Postman screenshot showing a GET request configuration. The URL is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/FB/a1b1db08-b881-44d9-b5ce-9c8880c71314/{{internalConsentId}}/transactions?bookingStatus=booked&dateFrom=2023-01-01&dateTo=2023-02-08`. The 'Tests' tab is selected, containing the following pre-request script:

```

1 const responseJson = pm.response.json();
2 let next = responseJson.payload.transactions.links;
3 if (next != null) {
4   postman.setEnvironmentVariable("next_page", next.next.href);
5 }
6 else {
7   postman.setEnvironmentVariable("next_page", null);
8 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO33PIRB42191A4819001000"
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "201074847",
          "entryReference": null,
          "endToEndId": null,
          "bookingDate": "2018-05-10",
          "valueDate": "2018-05-10",
          "executionDateTime": null,
          "transactionAmount": {
            "currency": "RON",
            "amount": 56.0
          },
          "transactionType": null,
          "exchangeRate": [],
          "creditorName": "POPESCU ION",
          "creditorAccount": {
            "iban": "RO33PIRB42191A4819001000"
          },
          "debtorName": null,
          "debtorAccount": {
            "iban": null
          },
        }
      ]
    }
  }
}
```

```

    "remittanceInformationUnstructured": "Internet Transfer to POP
ESCU ION",
    "proprietaryBankTransactionCode": null,
    "additionalInformation": null,
    "transactionStatus": null,
    "creditDebitIndicator": null
},
{
    "transactionId": "201074892",
    "entryReference": null,
    "endToEndId": null,
    "bookingDate": "2018-05-10",
    "valueDate": "2018-05-10",
    "executionDateTime": null,
    "transactionAmount": {
        "currency": "RON",
        "amount": 11.0
    },
    "transactionType": null,
    "exchangeRate": [],
    "creditorName": "POPESCU ION",
    "creditorAccount": {
        "iban": "RO33PIRB42191A4819001000"
    },
    "debtorName": null,
    "debtorAccount": {
        "iban": null
    },
    "remittanceInformationUnstructured": "Trs To: Popescu Ion",
    "proprietaryBankTransactionCode": null,
    "additionalInformation": null,
    "transactionStatus": null,
    "creditDebitIndicator": null
}
],
"pending": [],
"links": null
},
"balances": []
}
}

```

8. GetAccountDetails

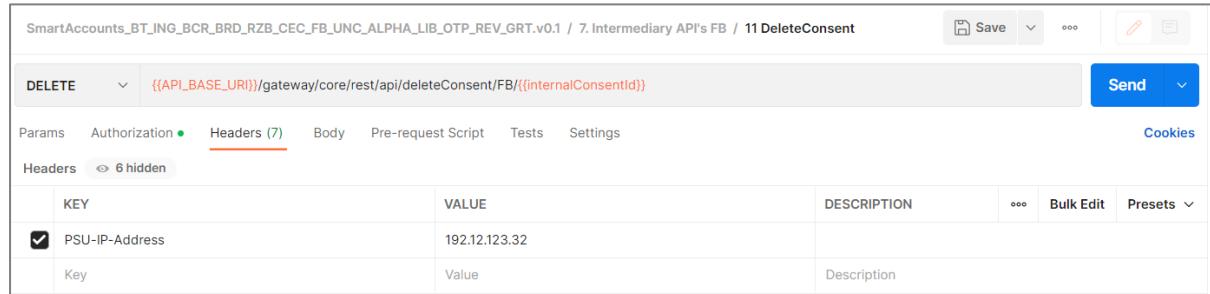
The screenshot shows the Postman application interface. The URL bar contains the endpoint: `({API_BASE_URI})/gateway/core/rest/api/accounts/FB/a1b1db08-b881-44d9-b5ce-9c8880c71314/({internalConsentId})/details`. Below the URL, there are tabs for Params, Authorization, Headers (7), Body, Pre-request Script, Tests, and Settings. The Headers tab is selected, showing a table with one row. The row has columns for KEY, VALUE, DESCRIPTION, and Presets. The KEY column contains 'PSU-IP-Address' with a checked checkbox, and the VALUE column contains '192.12.123.32'. The Description column is empty. There are also 'Bulk Edit' and 'Presets' buttons at the bottom of the table.

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "resourceId": "alb1db08-b881-44d9-b5ce-9c8880c71314",  
        "iban": "RO33PIRB42191A4819001000",  
        "currency": "RON",  
        "name": "Popescu Ion",  
        "product": null,  
        "cashAccountType": "*",  
        "status": null,  
        "bic": null,  
        "accountOfficerDescription": null,  
        "openingDate": null,  
        "customerSector": null,  
        "customerType": null,  
        "accountUsage": null,  
        "workingBalance": null,  
        "onlineActualBalance": null,  
        "balances": [],  
        "sortCodeAccountNumber": null  
    }  
}
```

9. DeleteConsent



The screenshot shows the Postman interface with the following details:

- Request Method: **DELETE**
- URL: `{{API_BASE_URI}}/gateway/core/rest/api/deleteConsent/FB/{{internalConsentId}}`
- Headers tab selected, showing:
 - Header Name: KEY
 - Header Value: VALUE
 - Description: Description
- Header Data:

KEY	VALUE	DESCRIPTION
PSU-IP-Address	192.12.123.32	Description
Key	Value	Description
- Buttons: Save, Send, Cookies

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "message": "Consent has been successfully revoked"  
    }  
}
```

10. RefreshToken

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Headers (9)

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> internalConsentId	{{internalConsentId}}			
<input checked="" type="checkbox"/> access_token	{{access_token}}			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Body (9)

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	{{refresh_token}}		
<input checked="" type="checkbox"/> client_id	{{client_id-partner}}		
Key	Value	Description	

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 7. Intermediary API's FB / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Tests (9)

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);
    
```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116311,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU4NjI1MDUsImlh
CI6MTY3NTg2MjIwNSwianRpIjoiM2MzMJDlZWQtNmI4Yy00ODgyLWI2ZDAtMjMwNWVjNmUwZGQyIiwi
aXNzIjoiHR0cDovL2tleWNsb2FrLWFwcClzZXJ2aNW1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW
50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTiymC11LOWE0LTQ2M2ItYWVmZi1mOTc4MTU2NDc3
ZDMiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYjk4YTEw
ZDAtNzBlNi00NDJkLWI0YWItMTZhMDZjNmM4YMu5IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6Wy
IvKjdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2Nlc3MiLCJ1bWFfYXV0aG9yaXphd
GlvbijdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjd
GlvbijdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtYW5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxp
bmtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUiOiJzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW
06MTE2
}
```

```

MzExIiwic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzExIiwiY2xpZW50SWQiOjJ0ZXN
0Y2xpZW50djEiLCJjbGllbnRIBn3N0IjoiMTAuMC4xNzYuMTg4IiwiZW1haWxfdmVyaWZpZWQiOmZhbHNlLC
JwcmVmZXJyZWRFdXN1cm5hbWUiOiJzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.Sy15ncbMa_-
IoJGfR4h5J47TVr5y1XX_NGUBgWh3Y0ukIzb2n3ZARhRXNvN_0g91IPx1A4STcTk2J_pWi4t6BRxaA6uLt
XMct5_OZJW4qWPUzSi15Fbjny3ow9M3pUGrhzW0dZgnU160yJs0dQ87SvyT76T53fd9BavCtgvLjpmgVRgr
8cGyk-EiRT4Vb3MB2jx_y3tqWD_M0XbbyZG9HGC4eUhV-gNuWizH0Y50VSMWothGKfWXSm6aKbA-
nFkVpAtBaIiVbSzGy2vwM-
AAuS97gzph65tNUybe8kitQ7vh4r8IDLpeHgcUMy3eojOFvF63jNXglGkEN7Ziw9bSg",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwa2lkIia6IC130WNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM2MzcyOTIisImhdCI6MTY3N
Tg2MjIwNSwianRpIjoiZmY3NDM2MTgtNzNjMy00ODZmLT1jMDAtMmE0Y2R1YzFlODE5IiwiXNzIjoiAHRO
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTIyMC11OWE0LTQ2M2ItYWFmZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOiJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImI5OGExMGQwLTcwZTYtNDQyZC1iNGFilTE2
YTA2YzzjOGJ1OSIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjMxMSJ9.Q1
vQ5wrM6sVbUqWZ0Hhd5M1spG1SIEbSgEFlQFvm6ZQ"
}
}

```

In case the "accessToken" expires, the response will have the following structure:

```
{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}
```

9. UniCredit Bank - Flow

1. Authenticate

The screenshot shows the Postman interface with the following details:

- Request URL:** {{MTLS_BASE_URL}}/gateway/authenticate/rest/api/token
- Method:** POST
- Body:** (form-data)

KEY	VALUE	DESCRIPTION	Bulk Edit
client_id	{{{client_id-partner}}}		
Key	Value	Description	
- Headers:** (9) (Pre-request Script, Tests, Settings)
- Cookies:**

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116313,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU4NjQ0MzEsImlh
CI6MTY3NTg2NDEzMswianRpIjoiOWM4OGY3MTQtZTk1OS00OWUzLWF1YzYtMWVjY2FmNWZjZjdIIiwiaXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfyzXiiLCJhenAiOj0ZN0Y2xpZW50djEiLCJzZXNzaWuX3N0YXR1IjoiZjg5MmZ1MWQtM
DgzzS00Zjg4LTgyM2QtZmM1YmE5MDI2ODE0IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFBsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYw5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidm11dy1wcm9maWx1I119fSwic2NvcGUIoijzbWFydGFjY291bnRzX2NvbnN1bnRfaWRfY2xhaW06MTE2
MzEzIiwic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltijoimTE2MzEzIiwiy2xpZW50SWQioij0ZN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xNzYuMTg4IiwiZ1haWxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOjzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.YXhjvY3fcaNT6LWLXdhbUBXo2zQ8Pvr-
n5kJxrS8fZOLF9Q6WpFH5c9gxHUC90-nNwC9US31UHF1XKFSWTqITE_1uv3rK-mqEvGU-
4SqvzbZ8jjw0IY6678I8vT0d2L8kPgCoXbs2lscCch5uT-
mteW1Jg4gawmwS51LzK_uJw9re9krtLWGif3qQdbhURN5WpNIf_Xx-
qz4uv5HxqsomsEN2Dm4FCmiqluhy5xJcjGA5SXRA8KUZkxkOejF08d-7K3XurF_VotD_17aR-
Vj25IN8myYZ4USar8jjwy7T6auDFXVp6A5Yx9oXi3FFzo02QqNTgI9ZUxuydjOeIV84w",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhY
2RmMi1kMjk3LTQ0YWItyjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM2NDAxMzEsImlh
CI6MTY3NTg2NDEzMswianRpIjoiZWIwYjRhODYtMjc0OS00YWJmLTk4MTItNTE5MjViZDRhYmM0Iiwi
aXNzIjoiiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImY4OTJmZTFkLTA4M2UtNGY4OC04MjNkLWZj
NWJhOTAyNjgxNCIsInNjb3BlIjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjMxMyJ9.ml
ISiVIJkKaaHIe48lmKsYSaLvb152kAHrfefTz6uLj4"
  }
}

```

2. initConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 8. Intermediary API's UNC / 4. InitConsent

POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}

Params Authorization Headers (11) Body Pre-request Script Tests Settings Cookies

Headers 8 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Content-Type	application/json				
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
<input checked="" type="checkbox"/> PSU-ID-Type	ALL				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 8. Intermediary API's UNC / 4. InitConsent

POST {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}

Params Authorization Headers (11) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

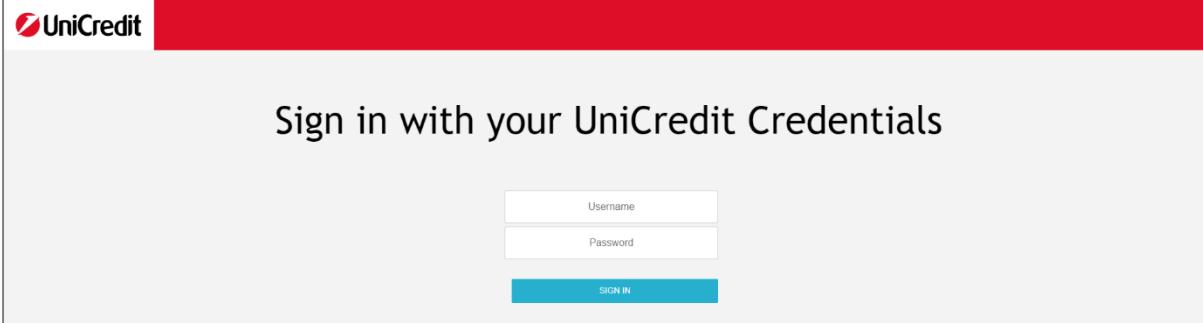
1
2
3
4
5
6
7
8
      "codeBank": "UNC",
      "periodOfValidity": 90,
      "psuEmail": "aaa.aaa@itsmartsystems.eu",
      "psuIntermediaryId": "324567",
      "redirectURL": "https://google.com",
      "TCaccepted": true
    ]
  ]
}

```

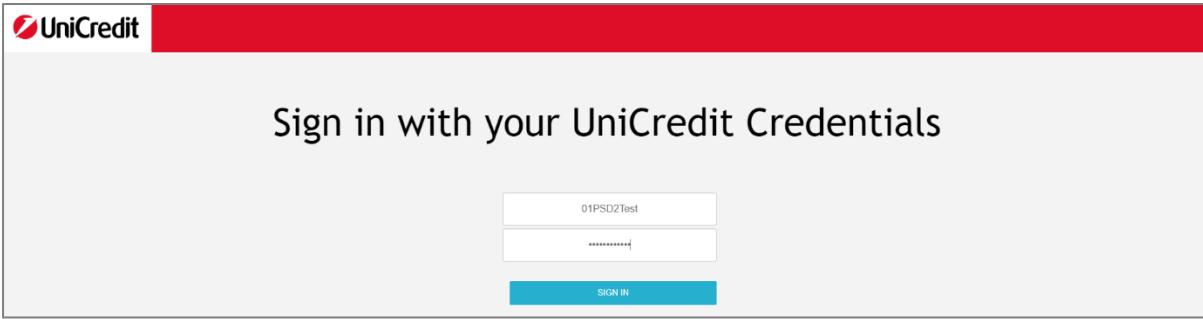
Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://authorization.api-sandbox.unicredit.eu:8403/sandbox/psd2/bg/hydrogen/h014/loginPSD2_BG.html?consentId=2163b89f-de7e-422d-a074-6f75b65077bd&correlationId=B3L1GB&authorisationId=2163b89f-de7e-422d-a074-6f75b65077bd&country=RO&state=ZXlKbGJtTWlPaUpCTVRJNFEwSkRMVWhUTWpVMklpd21ZV3huSWpvaVpHbH1JbjAuLm4zdWNiVk8wbkw3UWpHM1JLTk1VVkEuRmJkZXFGc09GTGhSLThTazRLbFlNdTFwRE1MRD1JTWRtOUFTRkFMNDBYb2RRZkM1NkRra25Rb1hyOE5zZ00zUjv5WDQyMnF5Y0E0ZDV0bEI2NXJMbm1hNjBYbGlqNm1ZR2tLcXN0TEN4TFpYNHBmb2czZ3ZxRTg5TFZ6UU9kZ1QubVpoel91WjhlcGRvZnFMX1M0WXo3UQ=="
      }
    }
  }
}
```

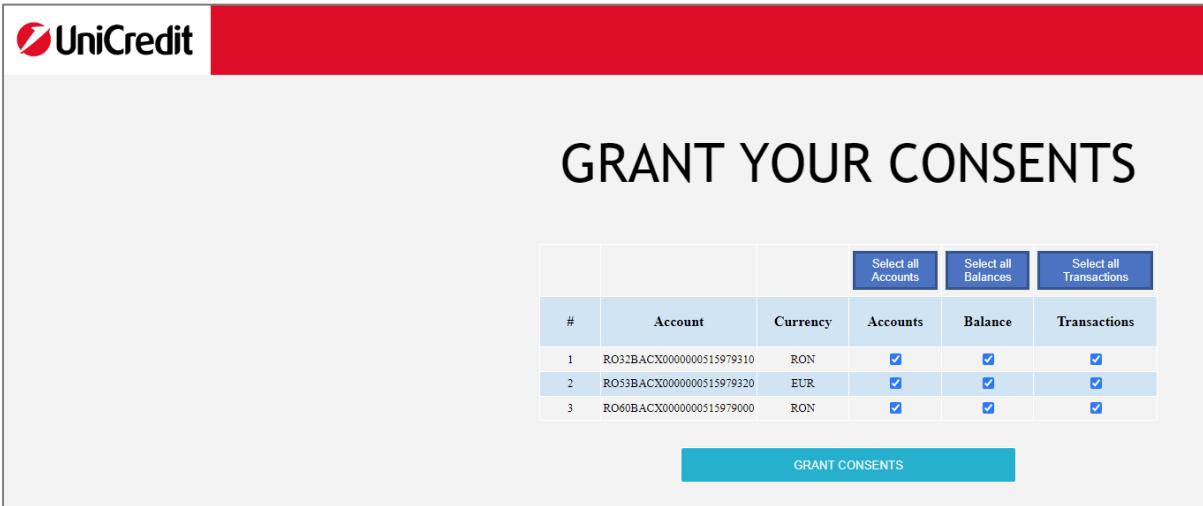
2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application



The screenshot shows the UniCredit sign-in page. At the top left is the UniCredit logo. The main heading is "Sign in with your UniCredit Credentials". Below it are two input fields: "Username" and "Password", both currently empty. A blue "SIGN IN" button is positioned at the bottom right of the input area.



This screenshot shows the same UniCredit sign-in page as above, but with the "Username" field populated with "01PSD2Test" and the "Password" field populated with "*****". The "SIGN IN" button remains at the bottom.



The screenshot shows the "GRANT YOUR CONSENTS" page. At the top left is the UniCredit logo. The main heading is "GRANT YOUR CONSENTS". Below it is a table with six columns: "#", "Account", "Currency", "Accounts", "Balance", and "Transactions". The first three columns have rows for accounts 1, 2, and 3. The last three columns each have a "Select all" checkbox at the top, followed by individual checkboxes for each account row. A blue "GRANT CONSENTS" button is located at the bottom right of the table area.

#	Account	Currency	Accounts	Balance	Transactions
1	RO32BACX0000000515979310	RON	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	RO33BACX0000000515979320	EUR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	RO60BACX0000000515979000	RON	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

After selecting at least one account and type of consent, and pressing "GRANT CONSENTS", thus authorising the consent based on granularity (which account, balances and transactions), developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application

 **UniCredit Bank**

ROMÂNĂ ✓

Bine ai venit!

Nume utilizator

Introduceți numele de utilizator

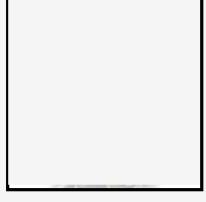
Această valoare este necesară

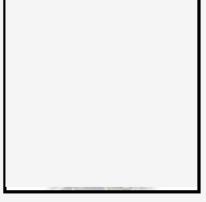
CONTINUĂ

 **UniCredit Bank**

SELECTEAZĂ IMAGINEA DE SECURITATE

Accesezi aplicația de pe un browser sau un computer nou, așa că te rugăm să ne confirmi imaginea de securitate. Îți reamintim că imaginea de securitate apare în meniu de setări, tab-ul de securitate.









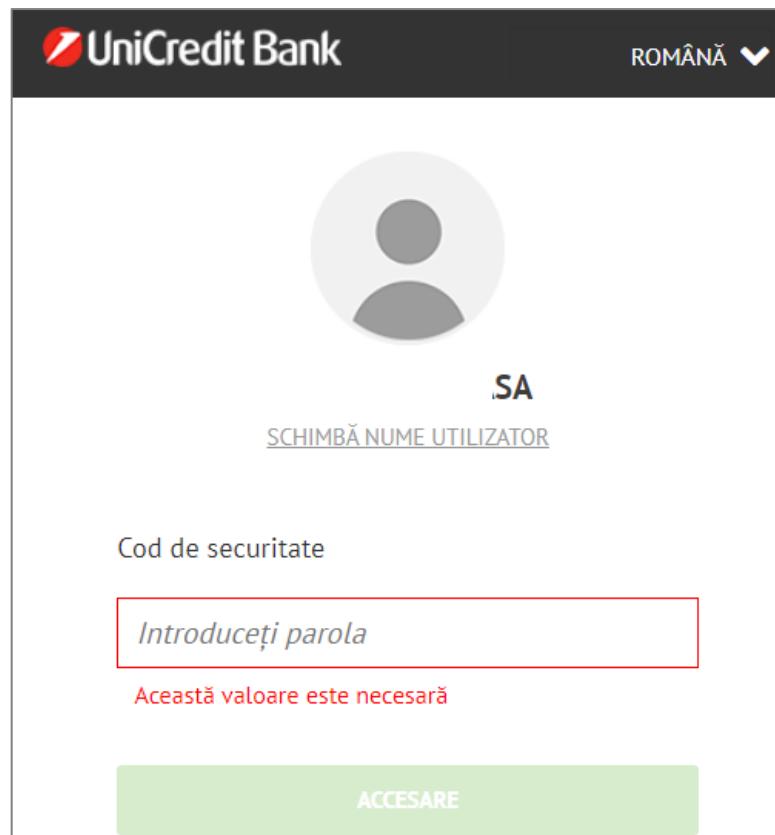
SELECTEAZĂ

Imaginea de securitate pe care o alegi va întări securitatea aplicației Online Banking. Mai multe detalii găsiți în acest document.

Dacă întâmpini probleme, te rugăm să ne contactezi la:

Telefon
+40 21 200 2020 (apel tarif normal în rețeaua fixă Telekom Romania)
(interior 1116 or 11232)

E-mail
BusinessNet: businessnet@unicredit.ro



Account Information Service Consent

Consent ID	
Current status	Received
Expiration date	01.05.2023
TPP Name	SMART FINTECH S.R.L.
Country registration number	
Country of registration	RO
TPP National Competent Authority	NBR
Permissions and accounts <small>?</small>	
<ul style="list-style-type: none">• Account Details <input type="checkbox"/> <input checked="" type="checkbox"/> Select all 08 1002 SAVINGS ACCOUNT RON 08 1000 CURRENT ACCOUNT RON• Account Balances <input type="checkbox"/> <input checked="" type="checkbox"/> Select all 02 SAVINGS ACCOUNT RON 00 CURRENT ACCOUNT RON• Account Transactions <input type="checkbox"/> <input checked="" type="checkbox"/> Select all SAVINGS ACCOUNT RON CURRENT ACCOUNT RON	
<small>With this new consent any other consent offered to the same account information service provider, if it</small>	

After selecting at least one account and type of consent, and authorising the consent based on granularity (which account, balances and transactions), PSU will be redirected to the established page.

3. GetConsent

The screenshot shows the Postman interface with a GET request to `((API_BASE_URI))/gateway/core/rest/api/consent/UNC/{internalConsentId}`. The Headers tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
<input checked="" type="checkbox"/> Content-Type	application/json				
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO32BACX0000000515979310"
        },
        {
          "iban": "RO53BACX0000000515979320"
        },
        {
          "iban": "RO60BACX0000000515979000"
        }
      ],
      "balances": [
        {
          "iban": "RO32BACX0000000515979310"
        },
        {
          "iban": "RO53BACX0000000515979320"
        },
        {
          "iban": "RO60BACX0000000515979000"
        }
      ],
      "transactions": [
        {
          "iban": "RO32BACX0000000515979310"
        },
        {
          "iban": "RO53BACX0000000515979320"
        }
      ]
    }
  }
}
```

```
        },
        {
            "iban": "RO60BACX0000000515979000"
        }
    ],
    "availableAccounts": null
},
"consentStatus": "valid",
"frequencyPerDay": 4,
"lastActionDate": "2023-02-08",
"validUntil": "2023-05-08",
"recurringIndicator": true,
"combinedServiceIndicator": null,
"creationDate": null
}
}
```

4. GetConsentStatus

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT... / 8. Intermediary API's U... / 6. GetConsentStat...

Save

GET [`\(\(API_BASE_URI\)\)/gateway/core/rest/api/consent/UNC/\({internalConsentId}\)/status`](#)

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Headers

<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32	
Key	Value	Description

Response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": {  
    "consentStatus": "valid"  
  }  
}
```

5. GetAccounts

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 8. Intermediary API's UNC / 7. GetAccounts

Save

GET [{{API_BASE_URI}}/gateway/core/rest/api/accounts/UNC/{{internalConsentId}}](#) Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Headers 6 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": [  
        {  
            "resourceId": "RO32BACX0000000515979310",  
            "iban": "RO32BACX0000000515979310",  
            "currency": "RON",  
            "name": "RON Acc for 01PSD2Test",  
            "product": "RBN",  
            "cashAccountType": "CACC",  
            "status": "enabled",  
            "bic": "BACXROBUXXX",  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        },  
        {  
            "resourceId": "RO53BACX0000000515979320",  
            "iban": "RO53BACX0000000515979320",  
            "currency": "EUR",  
            "name": "EUR Acc for 01PSD2Test",  
            "product": "RBN",  
            "cashAccountType": "CACC",  
            "status": "blocked",  
            "bic": "BACXROBUXXX",  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        },  
        {  
            "resourceId": "RO60BACX0000000515979000",  
            "iban": "RO60BACX0000000515979000",  
            "currency": "RON",  
            "name": "RON Acc for 01PSD2Test",  
            "product": "RBN",  
            "cashAccountType": "CACC",  
            "status": "enabled",  
            "bic": "BACXROBUXXX",  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        }  
    ]  
}
```

```

        "bic": "BACXROBUXXX",
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/UNC/RO32BACX000000515979310/{{internalConsentId}}/balances
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - Key: Value

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO32BACX000000515979310",
            "currency": "RON"
        },
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 4500.16
                },
                "balanceType": "expected",
                "referenceDate": null,
                "creditLimitIncluded": "false",
                "creditDebitIndicator": null,
                "lastChangeDateTime": null,
                "lastCommittedTransaction": null,
                "transactionStatus": null
            }
        ]
    }
}

```

}

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 8. Intermediary API's UN... / 9. GetTransactio...

GET `({{API_BASE_URI}})/gateway/core/rest/api/accounts/UNC/RO32BACX0000000515979310/{{internalConsentId}}/transactions?bookingStatus=booked&d`

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> next	<code>({{next_page}})</code>				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 8. Intermediary API's UN... / 9. GetTransactio...

GET `({{API_BASE_URI}})/gateway/core/rest/api/accounts/UNC/RO32BACX0000000515979310/{{internalConsentId}}/transactions?bookingStatus=booked&d`

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> bookingStatus	booked			
<input checked="" type="checkbox"/> dateFrom	2022-08-18			
<input checked="" type="checkbox"/> dateTo	2022-08-18			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 8. Intermediary API's UN... / 9. GetTransactio...

GET `({{API_BASE_URI}})/gateway/core/rest/api/accounts/UNC/RO32BACX0000000515979310/{{internalConsentId}}/transactions?bookingStatus=booked&d`

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

```

1 const responseJson = pm.response.json();
2 let next = responseJson.payload.transactions.links;
3 if (next != null) {
4   postman.setEnvironmentVariable("next_page", next.next.href);
5 }
6 else {
7   postman.setEnvironmentVariable("next_page", null);
8 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO32BACX0000000515979310",
      "currency": "RON"
    },
    "transactions": {

```

```

        "booked": [],
        "pending": [],
        "links": null
    },
    "balances": []
}
}

```

8. GetAccountDetails

The screenshot shows the Postman interface with a GET request to `/accounts/RO32BACX000000515979310/({internalConsentId})/details`. The Headers tab is selected, displaying the following configuration:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "resourceId": "RO32BACX000000515979310",
        "iban": "RO32BACX000000515979310",
        "currency": "RON",
        "name": "RON Acc for 01PSD2Test",
        "product": "RBN",
        "cashAccountType": "CACC",
        "status": "enabled",
        "bic": "BACXROBUXXX",
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}

```

9. DeleteConsent

The screenshot shows a Postman request configuration for the 'DeleteConsent' API endpoint. The method is set to 'DELETE' and the URL is `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/UNC/((internalConsentId))`. The 'Headers' tab is selected, showing two header entries: 'PSU-IP-Address' with value '192.12.123.32' and 'Key' with value 'Value'. Other tabs like 'Params', 'Body', and 'Tests' are visible at the top.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "message": "Consent has been successfully revoked"
  }
}
```

10. RefreshToken

The screenshot shows a Postman request configuration for the 'RefreshToken' API endpoint. The method is set to 'POST' and the URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The 'Headers' tab is selected, showing three header entries: 'internalConsentId' with value '{{internalConsentId}}', 'access_token' with value '{{access_token}}', and 'Key' with value 'Value'. Other tabs like 'Params', 'Body', and 'Tests' are visible at the top.

The screenshot shows a Postman request configuration for the 'RefreshToken' API endpoint. The method is set to 'POST' and the URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The 'Body' tab is selected, showing the 'form-data' option is chosen. The body contains two fields: 'refresh_token' with value '{{refresh_token}}' and 'client_id' with value '{{client_id-partner}}'. Other tabs like 'Params', 'Headers', and 'Tests' are visible at the top.

```

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken
Params Authorization Headers (9) Body • Pre-request Script Tests • Settings Cookies
1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116313,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC14alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU4Njc4MTIsImhd
CI6MTY3NTg2NzUxMiwianRpIjoizWVizjcwODMtNGM3MC00ZTF1LWI0zjUtMDM2NTRiYjF1YTNmIiwiaXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWiiOii2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfYXZXiIiLCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiZjg5MmZ1MWQtM
DgzzS00Zjg4LTgyM2QtZmM1YmE5MDI2ODE0IiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFBsbV9hY2N1c3MiOnsicm9sZXMiolsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdG1vbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2x1cyI6WyJ1bWFfcHJvdGVjdG1vbiJ
JdfSwiYWNjb3VudCI6eyJyb2x1cyI6WyJtyW5h2UtYWNjb3VudCiisIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidm1ldy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydGFjY291bnRzX2NvbnnlbnRfaWRfY2xhaW06MTE2
MzEzIiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzEzIiwiY2xpZW50SWQiOj0ZXN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xNzYuMTg4IiwiZWh1haWxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbuUiOjzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QRWkc
mVzcyI6IjewLjAuMTc2LjE4OCJ9.QHSe9CJN16QiSqsW5iG5bj_NAOglmz85W61i84mBd8Ka2cnqKjkB6qc
RWVrXJKECDMlgX3CTqws92b8vXtv8giGh3C6EQohfIPFefILdq-CRqnomYnRUzzJaW0CY1-mzSlR-
eGnQIKQHRM9kAEacEHv-
Ei1Uk8Lf1btRNrnFK8f1zUohGIxYu6xurvDIQt1anF0M11nMTZkod8JVnz2tnoGH8m5VBj7RvDTEOcL72J6
gUNRXcZ9i5HbeDBReNaHY1fkA-eoVRUejTPma-1F8_KZgdJWrWDY1Wp-
6odoiqjoxK1q3rNuMWNw2xPvMeX1Ikciakc0R41nD7DMFAAQ5",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC130WNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NtC0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM2NDAxMzEsImhdCI6MTY3N
Tg2NzUxMiwianRpIjoizNzEyNWJ1ZTQtOTM4My00ZTYwLWJhMjktMmNhNjJiZjI1MmZjIiwiXNzIjoiahr0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpWIjoiGVzdGNsaWVudHYxIiwiC2Vzc21vb19zdGF0ZSI6ImY4OTJmZTFkLTA4M2UtNGY4OC04MjNkLWZj
NWJhOTAyNjgxNCIsInNjb3BlIjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjMxMyJ9.J0
DLf5gDSvbEGeRbY86PzK3KxPygh0XizDskhNzob-0"
  }
}
```

In case the "accessToken" expires, the response will have the following structure:

```
{
  "status": 401,
  "messageStatus": "Unauthorized",
}
```

```

    "moreDetails": {
        "text": "Invalid access token"
    }
}

```

10. Revolut - Flow

1. Authenticate

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 1. Authenticate

POST `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/token`

Params Authorization Headers (9) Body **Pre-request Script** Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> client_id	<code>{{client_id-partner}}</code>			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 1. Authenticate

POST `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/token`

Params Authorization Headers (8) Body **Pre-request Script** Tests Settings Cookies

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 116330,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0eyJleHAIoJE2NzU4Njg0NDAsImIhdCI6MTY3NTg2ODE0MCwianRpIjoimDI0OWU0NGItMzFjYi00YTQ1LTk1YmQtZTljm2QwN2EzNWm3IiwiiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQilCJzdWIiOiI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOiJCZWfYzXiilCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiOTQ4YWE2MDgtOTQ4OC00YjJlLWE0NzgtN2U4NTYyMjkzDgwIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKijdLCJyZWFBsV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtYW5hZ2UtYWNjb3VudCIsImIhbmcFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWxli119fSwic2NvcGUiOiJzbWFydGFjY291bnRzX2NvbnN1bnRfaWRFy2xhaW06MTE2MzMwIiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzMwIiwiY2xpZW50SWQiOiJ0ZXN0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xNzYuMTg4IiwiZ1haWxfdmVyaWZpZWQiOmZhbHN1LCJwcmVmZXJyZWRfdXN1cm5hbWUiOiJzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkcMvzcyI6IjEwLjAuMTc2LjE4OCJ9.OiJIYpJrISQOBvOR2GXCbGGN_YiIS-
```

```

320tyjYeSF1OKxTPvGfLi_LQ1FjhZfKMsGNDrof-Ui4OQ-v-
6uhtrZUxTkKP8667ZgWgy41VOq7EH7A4rLwLZCiJGaA0GWQxAiHdik1gWMFp1sYRoUv_8epOopLv88qtCCI
lQoK-JfR_HXAh5H2PFarpGtezugTaISRONP9Qct11AnwyzIrmxxLZ--
Bu4ormk8Mr9PH41KNpRzgTQstTqHyzhoC4VZJMM0c0sxmwdoCrqBtcIO1rOQO2EibH0oFdd0aRyY1BoYp31
ekqsHUw4bXyTx-fiYMCYMPYbKaJ9TSnOyufKV69PMtg",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwa2lkIia6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NtC0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM2NDQxNDAsImhdCI6MTY3N
Tg2ODE0MCwianRpIjoiM2I2NGY5NWYtMDc5Zi00Mjk1LTgwNTItZmEzNzEzOGFhZjAyIiwaXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN10jgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZT1yMC11OWE0LTQ2M2ItYWFmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpWIjoidGVzdGNsaWvdHYxIiwigc2Vzc2lvb19zdGF0ZSI6Ijk0OGFhNjA4LTk0ODgtNGIyZS1hNDc4LTd1
ODU2MjI5M2Q4MCIsInNjb3B1Ijoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYwltojExNjMzMCJ9.26
9s7uELve8ERs9qaB5weWiHEEyfw15J0kRkEwEJjBU"
}
}

```

2. initConsent

The screenshot shows the Postman interface with the following details:

- Request URL:** {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}
- Method:** POST
- Headers:**
 - Content-Type: application/json
 - PSU-IP-Address: 192.12.110.32
 - PSU-ID-Type: ALL
 - Country: UK
 - Key: Value
- Body:** (Empty)
- Tests:** (Empty)
- Settings:** (Empty)

The screenshot shows the Postman interface with the following details:

- Request URL:** {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}
- Method:** POST
- Headers:**
 - Content-Type: application/json
- Body:**

```

1   "codeBank": "REV",
2   "periodOfValid": 90,
3   "psuEmail": "aaa.aaaa@itsmartsystems.eu",
4   "psuIntermediaId": "324567",
5   "redirectURL": "https://google.com",
6   "TCaccepted": true
7
8

```
- Tests:** (Empty)
- Settings:** (Empty)

Response:

```

{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {

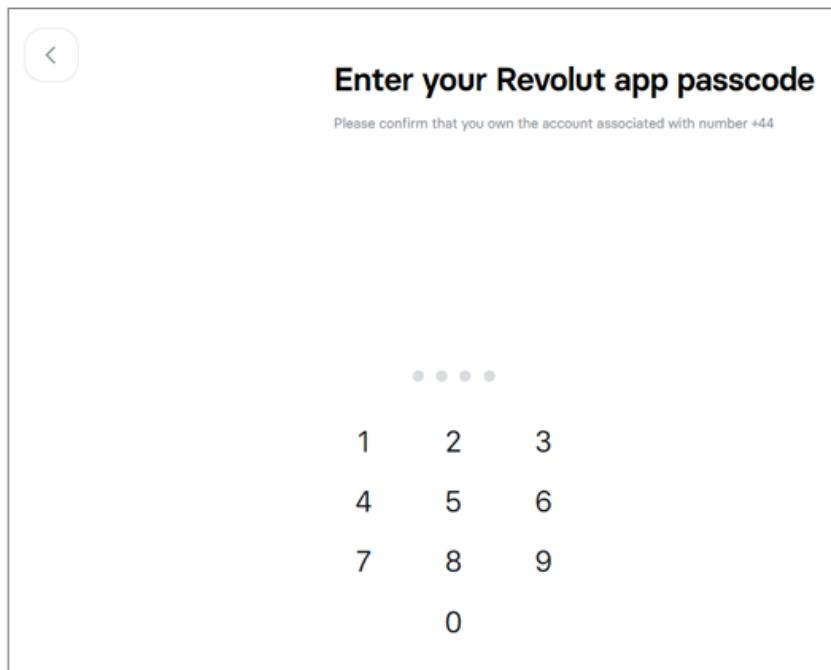
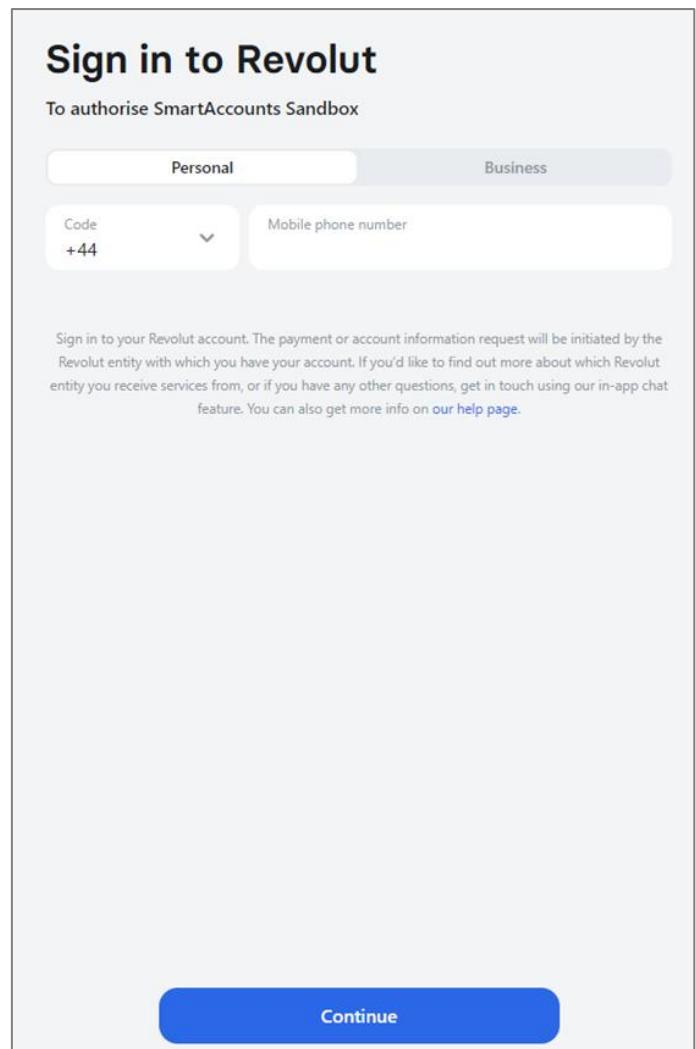
```

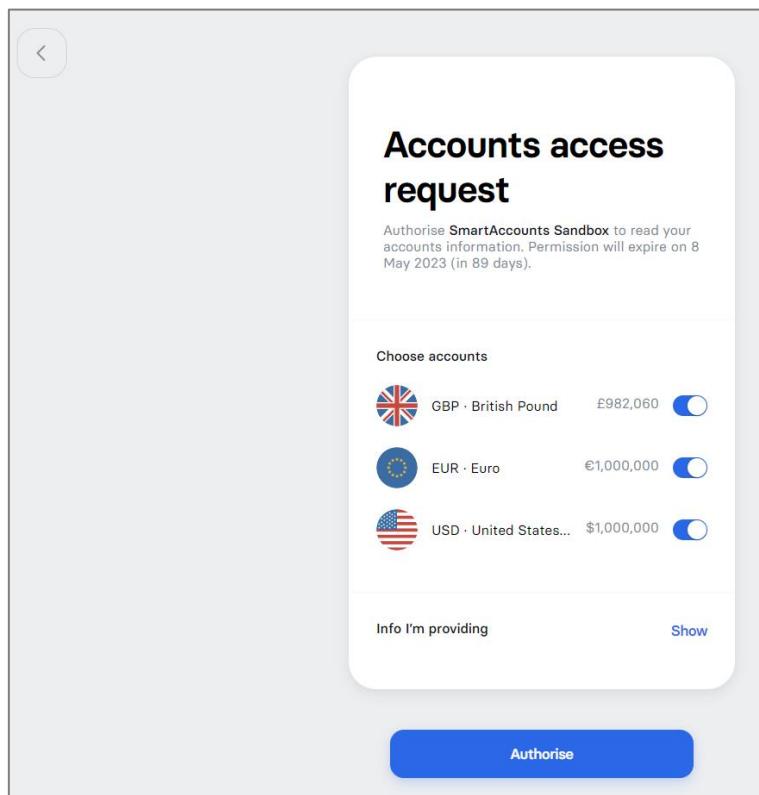
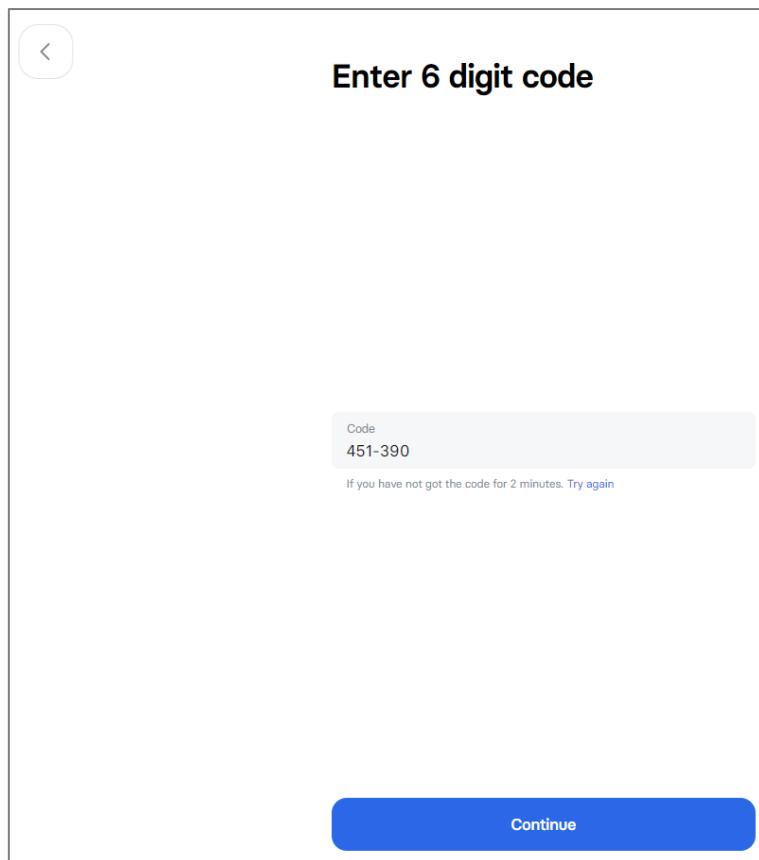
```

    "scaOAuth": {
        "href": "https://sandbox-
oba.revolut.com/ui/index.html?state=eyJlbmMiOiJBMTI4Q0JDLUhTMjU2IiwiYWxnIjoizGlyIiw
iY3R5IjoiSldUIn0..g7Y7KAqK-
v4TAqnLHizasg.tv06Wks9Fits0SpfyDPovk_2kghSVpyQyIEtC8zbklRylgdqs707n_tVd_SU10Vdy8tpT
d0JD2WneN9eF5skqD_eJWZ1vZbjtVwilmeey2Xe1azccLlcj9wBaQ80tb1nwcesZVVXiZmKyW2-
Cf1Q2EUVO9pc2Ci6hU9k8Y19B0hsbTqffeNp0WzoXS7rh9p0kE-oLelezICH7oZB4SF-gOacn-
i2m0_wY7hGVmWqfLOU9CPwC9PN4zqfY5cfFKT-
OZkE1DATRr19zjKDS0Pn0IdoRWSYsAjduNgjYaVUW6PZSz0DeyVjIQIAmP0hMncya8SCoNRU5KiMFpSF1Y5
NVdMZejrqx250wdk0XohnIwUwinW0vpzKqdWsQahePvoAt1Yq6lnJKpUwCy3xZqfXZiBuMCUHSpprjH8dWp
5MoOnSq7Yw0ykKTSPuFPwrPasrl1NlllVMElwMYm4Tn45Gj82xK_T4hCb1EZRCUDK1xKYuUK6kg-
FULclLU6ihmTYoQNF7hcTXPbzPKM81H1MDjt3u6TDaiuLppcJRph8vMUqS3sg5LpGLSR9OEFL_UbeZKcjP
cTSxR76yftjK1-NEv8bHfzxwLzTbkW--weN_kh3JITzTzo08UzdA4dzzox-
GFVvxOpOyCKXg5q30L2KntF8KUd0Fy_zGQaEvKvjEujSrbSQW8UM4pa8Zhs9BxH1h5bd8KnILVPW1WaZ9i
Y4uG8ecTBQb1fIMztRAcDMgOS73uffF3nEwVSe2J6C264nyfOEbvew_LMEH9k2320SYAa9d1bXh4IF_VBPg8
qD-P5RRPvd839SdQHUbjOxTwIr7ZDbW-Eon9s5R4v-
5TPQrrqoToVJxqbbE6FS6qBhf2A3yCaZ72YjJmAEeupKesTOBSEGWi-SP3PMuUAZoNyB9u3BQ8tterj-
p0alQ9V51MyCy7I2CF5t4FRQnvmMIaBNoj6WdmkwDadF9t86PE06gtubwkzTDRt4x5kZyRhFUj4VqmJrcPZ
CTnsAqPibmFk0EF2kj_b_j9G8wDGHbTsTXiBaOJRe1G4hw0ruLNomWsbkoXYYbDkWYibk5QvpmYjk3L.spkU
X0MaD91f_fj1CsgPTQ&response_type=code%20id_token&client_id=224062b5-7bb5-45ce-8118-
c8900851bcae&redirect_uri=https://api.sandboxaccounts.smartfintech.eu/gateway/core/
rest/api/confirmation&scope=openid%20accounts&nonce=n-
0S6_WzA2Mj&request=eyJhbGciOiJQUzI1NiIsInR5cCI6IkpxVCIsImtpZCI6IjM3MzA3NDIwLWRkMGmt
NGQzMylMznjLTM0ZTU2YWUyOWI0NCj9.eyJhdWQiOiIyMjQwNjJiNS03YmI1LTQ1Y2UtODExOC1jODkwMD
g1MWJjYWUiLCJpc3MiOiJzYW5kYm94Lw1zcyIsInJlc3Bvbnn1X3R5cGUioIjb2R1IG1kX3Rva2VuIwiY
2xpZW50X2lkIjoiMjI0MDYyYjUtN2JiNS00NWN1LTgxMTgtYzg5MDA4NTFiY2FlIwicmVkaXJ1Y3RfdXjP
IjoiaHR0cHM6Ly9hcGkuc2FuZGJveGFjY291bnRzLnNtYXJ0ZmludGVjaC51ds9nYXR1d2F5L2NvcnUvcmV
zdc9hcGkvY29uZmlybWF0aW9uIiwig2NvcGUioIjvcGVuaWQgYWNjb3VudHMiLCJjbGFpbXMiOnsiaWRfdG
9rzW4iOnsib3BlbmJhbmtpbmdfaW50ZW50X2lkIjp7InZhbHV1IjoiOGQxNDgyYzQtOGFiNC00MmFmLWJk0
DQtMjMwYTU4YTFmNmQwIiwiZXNzZW50aWFsIjp0cnVlfX19fQ.Kjfvb0xck8nGfJeHRXxf7s2UsRI-
CjrefbDmn3oLViaKjHNPPVqxF0yd81MmZzGA_eqpgKEbAxbZMwCE8E8cTvZW-
aEKXKa3IN9IsRFucxqMG6OhQjZR4Oe5iiwchWuFrBm4tGGVVCBqq9sECKz36MpnvXPg7pI-
4Jgb6DTJ87atpUtNp6XJRvx4qOF4IBbDQ9XG54wonIazxl8kRNBPGBcxc2Jwdp61Ha6LQeLizPcbNjCnxgU
hzLwMIL7WQdsQsIVIGuQ0o3iHi2_2tMge0ZoSppoBvlpnQu3BhEOF1j3VL8XoGcs_oz-
fYPhvkvnimksBv3aPYE1mGcab6WNF4A"
    }
}
}
}

```

2.1 initCosent in SANDBOX – the example below shows how the developer will interact with bank's internet banking application







Access granted

You will be now redirected back to SmartAccounts Sandbox

After selecting accounts and pressing "Authorise", thus authorising the consent for selected account(s) developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application

Sign in to Revolut

To authorise

Personal

Business



Switch to Personal account

Continue in Revolut >

Sign in to your Revolut account. The payment or account information request will be initiated by the Revolut entity with which you have your account. If you'd like to find out more about which Revolut entity you receive services from, or if you have any other questions, get in touch using our in-app chat feature. You can also get more info on [our help page](#).

Log in to Revolut

Phone number

Email



Phone number

Lost access to phone number

Continue

Don't have an account?

Sign up



Log in with QR code

Scan this code with your phone camera to log in instantly. Revolut app version 8.93 or higher is required

After inserting phone number, PSU will be redirected into a page where Revolut account passcode must be inserted (not card's PIN).

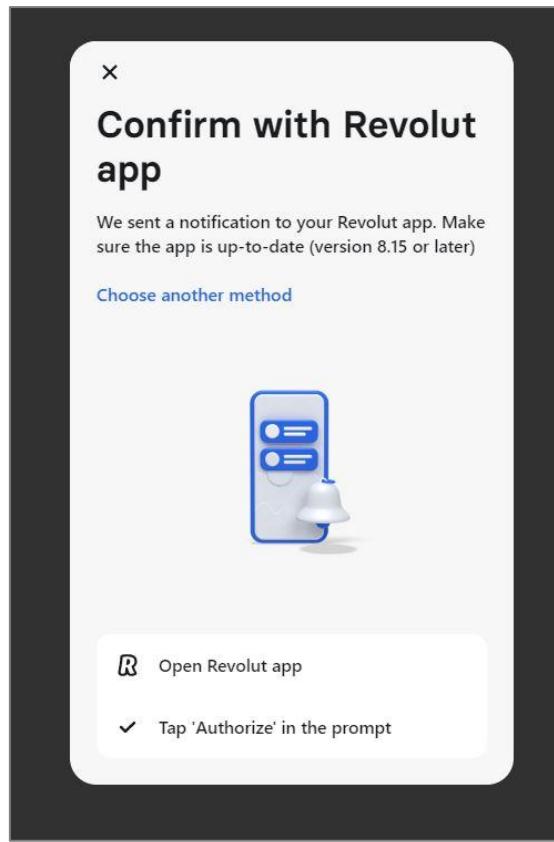
←
Enter passcode

....|

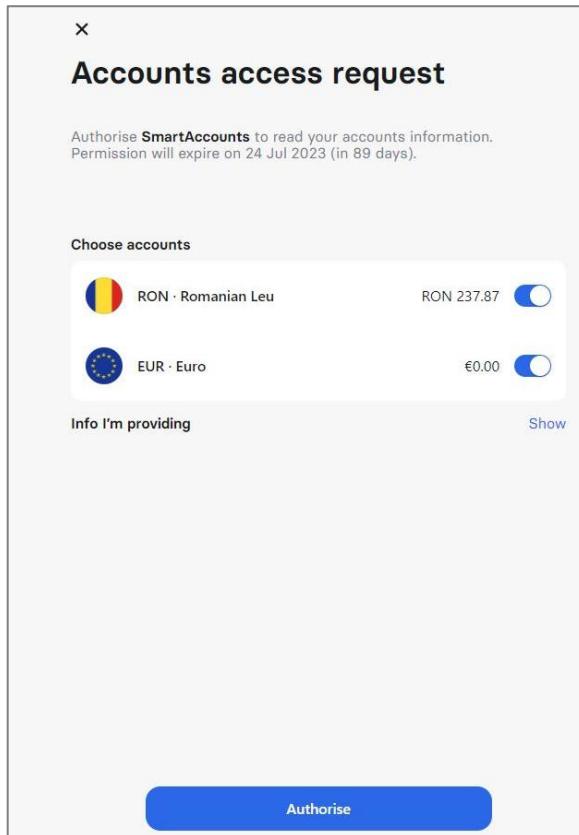
Only digits are allowed

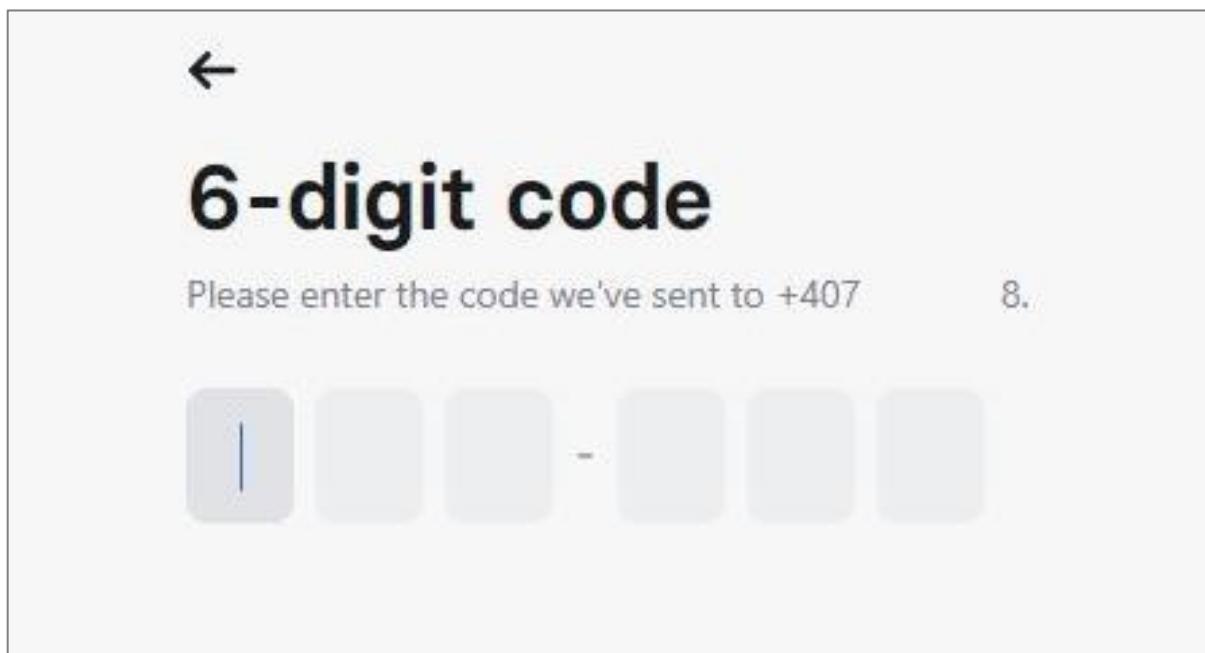
Forgot your passcode?

Continue



After tapping "Authorize" in the prompt from Revolut APP, PSU will be redirected into the web page below.





After inserting SMS code, PSU will be redirected to the established page.

3. GetConsent

A screenshot of the Postman API client interface. The header bar shows the project name "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 5. GetConsent". The main area shows a GET request with the URL: `{API_BASE_URI}/gateway/core/rest/api/consent/REV/{internalConsentId}`. The "Headers" tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32		
<input checked="" type="checkbox"/> Content-Type	application/json		
Key	Value	Description	

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "access": {
            "accounts": [],
            "balances": [],
            "transactions": [],
            "availableAccounts": null
        },
        "consentStatus": "valid",
        "frequencyPerDay": null,
        "lastActionDate": "2023-02-08",
        "validUntil": "2023-05-08",
        "recurringIndicator": null,
        "combinedServiceIndicator": null,
        "creationDate": null
    }
}
```

4. GetConsentStatus

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.... / 12. Intermediary API's R... / 6. GetConsentStat...

GET [\({{API_BASE_URI}}\)/gateway/core/rest/api/consent/REV/{{internalConsentId}}/status](https://{{API_BASE_URI}}/gateway/core/rest/api/consent/REV/{{internalConsentId}}/status) Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "valid"
    }
}
```

5. GetAccounts

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 7. GetAccounts

GET [\({{API_BASE_URI}}\)/gateway/core/rest/api/accounts/REV/{{internalConsentId}}](https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/REV/{{internalConsentId}}) Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": [  
        {  
            "resourceId": "b66af2e5-9dcf-4520-9fd2-12a451ae4309",  
            "iban": null,  
            "currency": "GBP",  
            "name": "Jan Smith",  
            "product": null,  
            "cashAccountType": "CurrentAccount",  
            "status": null,  
            "bic": null,  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": "60424981765417"  
        },  
        {  
            "resourceId": "1452fe59-464b-4aa4-9d8c-68172271bf7e",  
            "iban": "DK2377733760723902",  
            "currency": "EUR",  
            "name": "Jan Smith",  
            "product": null,  
            "cashAccountType": "CurrentAccount",  
            "status": null,  
            "bic": null,  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        },  
        {  
            "resourceId": "fb42d579-437e-4115-8829-40b0eca04001",  
            "iban": "DK6705166603207318",  
            "currency": "USD",  
            "name": "Jan Smith",  
            "product": null,  
            "cashAccountType": "CurrentAccount",  
            "status": null,  
            "bic": null,  
            "accountOfficerDescription": null,  
            "openingDate": null,  
            "customerSector": null,  
            "customerType": null,  
            "accountUsage": null,  
            "workingBalance": null,  
            "onlineActualBalance": null,  
            "balances": [],  
            "sortCodeAccountNumber": null  
        }  
    ]  
}
```

```

        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {},
    "balances": [
      {
        "balanceAmount": {
          "currency": "USD",
          "amount": 1000000.0
        },
        "balanceType": "InterimAvailable",
        "referenceDate": "2020-08-24",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null,
        "lastChangeDateTime": null,
        "lastCommittedTransaction": null,
        "transactionStatus": null
      }
    ]
  }
}
```

7. GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 12. Intermediary API's R... / 9. GetTransactio...

GET [{{internalConsentId}}](https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/REV/b66af2e5-9dcf-4520-9fd2-12a451ae4309/{{internalConsentId}}/transactions?bookingStatus=)

Send

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Headers (6 hidden)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
<input checked="" type="checkbox"/> next	<code>{{next_page}}</code>				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 12. Intermediary API's R... / 9. GetTransactio...

GET [{{internalConsentId}}](https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/REV/b66af2e5-9dcf-4520-9fd2-12a451ae4309/{{internalConsentId}}/transactions?bookingStatus=)

Send

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> bookingStatus	booked			
<input checked="" type="checkbox"/> dateFrom	2022-12-09			
<input checked="" type="checkbox"/> dateTo	2023-02-08			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 12. Intermediary API's R... / 9. GetTransactio...

GET [{{internalConsentId}}](https://{{API_BASE_URI}}/gateway/core/rest/api/accounts/REV/b66af2e5-9dcf-4520-9fd2-12a451ae4309/{{internalConsentId}}/transactions?bookingStatus=)

Send

Params • Authorization • Headers (8) Body Pre-request Script Tests • Settings Cookies

```

1 | const responseJson = pm.response.json();
2 | let next = responseJson.payload.transactions.links;
3 | if (next != null) {
4 |   postman.setEnvironmentVariable("next_page", next.next.href);
5 | }
6 | else {
7 |   postman.setEnvironmentVariable("next_page", null);
8 | }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {},
    "transactions": {
      "booked": [],
      "pending": [
        {
          "transactionId": "63b8295a-355e-a081-8b28-e645cc07a0d3",
          "entryReference": null,
          "endToEndId": null,

```

```

        "bookingDate": "2023-01-06",
        "valueDate": null,
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "GBP",
            "amount": 50.0
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": null,
        "creditorAccount": {
            "iban": null
        },
        "debtorName": null,
        "debtorAccount": {
            "iban": null
        },
        "remittanceInformationUnstructured": null,
        "proprietaryBankTransactionCode": "TRANSFER",
        "additionalInformation": "To ReceiverCo ReceiverRef",
        "transactionStatus": "Pending",
        "creditDebitIndicator": "Debit"
    }
],
"links": null
},
"balances": []
}
}

```

8. GetAccountDetails

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT...'. A GET request is defined with the URL `{{API_BASE_URI}}/gateway/core/rest/api/accounts/REV/b66af2e5-9dcf-4520-9fd2-12a451ae4309/({internalConsentId})/details`. The 'Headers' tab is active, displaying the following header:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "resourceId": "b66af2e5-9dcf-4520-9fd2-12a451ae4309",
        "iban": null,
        "currency": "GBP",
        "name": null,

```

```

        "product": null,
        "cashAccountType": "CurrentAccount",
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}

```

9. DeleteConsent

The screenshot shows a Postman request configuration. The method is set to **DELETE** and the URL is `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/REV/{{internalConsentId}}`. The **Headers** tab is selected, displaying a table with one row. The row has two columns: **KEY** and **VALUE**. The **KEY** column contains 'PSU-IP-Address' and the **VALUE** column contains '192.123.123.32'. There are also tabs for **Params**, **Authorization**, **Body**, **Pre-request Script**, **Tests**, and **Settings**.

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "message": "Consent has been successfully revoked"
    }
}
```

10. RefreshToken

The screenshot shows a Postman request configuration. The method is set to **POST** and the URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The **Headers** tab is selected, displaying a table with three rows. The first two rows have **KEY** columns containing 'internalConsentId' and 'access_token' respectively, and **VALUE** columns containing '{{internalConsentId}}' and '{{access_token}}'. The third row has a **Key** column and a **Value** column. There are also tabs for **Params**, **Authorization**, **Body**, **Pre-request Script**, **Tests**, and **Settings**.

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Params Authorization Headers (9) Body ● Pre-request Script Tests ● Settings Cookies

Body (form-data)

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	{{refresh_token}}			
<input checked="" type="checkbox"/> client_id	{{client_id-partner}}			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 12. Intermediary API's REV / 1.1 RefreshToken

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken

Params Authorization Headers (9) Body ● Pre-request Script Tests ● Settings Cookies

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);
    
```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116330,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhSzU0QndGYXFiMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOje2NzU4NzE0OTQsImlh
CI6MTY3NTg3MTE5NCwianRpIjoiNjIwNmJiMjAtYjzjYS00Y2M2LTljMDEtMTQ2ZmY4Mjg3NzUzIiwiaXNz
IjoiHR0cDovL2tleWNsb2FrLWFwcClzZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQjLCJzdWIiOiI2NmE2ZTlYMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfyzXiiLCJhenAiOjJ0ZXN0Y2xpZW50djEiLCjZXNzaWu9uX3N0YXR1IjoiOTQ4YWE2MDgtO
TQ4OC00YjJlWE0NzgtN2U4NTYyMjkzZDgwIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKijD
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiolsib2ZmbGluZv9hY2N1c3MiLCj1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VyzY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbi
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtyW5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTE2
MzMwIiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzMwIiwiY2xpZW50SWQiOjJ0ZXN
0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4xNzYuMTg4IiwiZWh1haWxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOjJzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.fn7zA1kJr8TViEQVItnvHiBkd3PArBnrmaCJQf_dBbu0gDqHItxGu0I
xh18-uqdZrsKcX2DNYy0go0ohdyugnEDHrbt6F_uQ5jdfZOUreIILQ4U-NObk5mdrjiirelf-
KemmgA2BccdwuUEbsuE5TB8IqlB3SeVb88U9J3nkiGFDDagxUQAeWdqyKjhWyeuTlob7aJjIvxmt7VMciJQ
DcxVwPyOtmgm40K9wV6LuwzTF4MlyP93-3SYux9wO1rYAYj-nntgRWEkoArxZMULnwi8b9K9-
I_PwI2w5dkBCC5CYbMglc6yJofGb93fHRH7qLac16eL1LJxcK9na7DrC47A",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl3OWNhY
2RmMi1kMjk3LTQ0YWIYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOje2ODM2NDQxNDAsImlhCI6MTY3N
Tg3MTE5NCwianRpIjoiMTg4Y2EwNjUtZG15OS00Y2E1LWE3NDgtMDQyN2QwNTZjMjAyIiwiXNzIjoiAHR0
cDovL2tleWNsb2FrLWFwcClzZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOjI2NmE2ZTlYMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW5lIiwiY
    
```

```

XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc2lvbl9zdGF0ZSI6Ijk0OGFhNjA4LTk0ODgtNGIyZS1hNDc4LTd1
ODU2MjI5M2Q4MCIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjExNjMzMCJ9.UO
Xt4a76Jm4w2dazBgQK-bfQfRg2r_OjSZFJ_GBlEVc"
}
}

```

In case the "accessToken" expires, the response will have the following structure:

```
{
  "status": 401,
  "messageStatus": "Unauthorized",
  "moreDetails": {
    "text": "Invalid access token"
  }
}
```

11. OTP Bank - Flow

1. Authenticate

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> client_id	{{client_id-partner}}			
Key	Value	Description		

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116346,
    "access_token": "eyJhbGciOiJSUzI1NiIiInR5cCIgOiAiSldeIiwiia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU5MjYxNjgsImhd
CI6MTY3NTkyNTg2OCwianRpIjoiZWYxMDFiMGItMzk5NC00Yz2LWEyNWEtNjBlNDZjNjNmZTA1IiwiAXNz
IjoiHR0cDovI2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyI

```

```

sImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTIyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOiJCZWfyzXiiLCJhenAiOij0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNWZkODd1ZGUtY
WNkNC00YTZkLTgyZGYtOWIyYzlhYTI0MzBjIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlslb2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbij
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbGllbnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbi
JdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtYW5hZ2UtYWNjb3VudCIsIm1hbmcFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTE2
MzQ2Iiwick1hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzQ2IiwiY2xpZW50SWQiOij0ZXN
0Y2xpZW50djEiLCJjbGllbnRIBn0IjoiMTAuMC4xNzYuMTg4IiwiZWhaWxfdmVyaWzpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOijzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.IEIZE4znoxD4n1keckkZcQdm0Sk53zh3j0p_LhE-
CNW8S6GtS7wFNMUja5tDBmhg_4fsXhUNzjWyDAU0kCZ_KRWvWHPBaC522gV9SRWbPy4hi5Yk5Dt-
P14FLCDju2-eY1hTmsLARAOnfs67WiPXaBIVH1A-ubSkXYzO-
iKPTd1FZx3ui8FRzlc1YIzZjDmNapaRBEdy9XnBSeufCO5btw3Xoqy8F820HFkkeyKGz2djTwiyhM3o10D
zrTGvu6ep8xyuWJoyAmiyrasd9Xp5aL7FYQ7EqJiAy8YSXG8G0KYdSOLzEP_-
t790DMzCkg8_OqyymyV7lIsvOsml8vQA",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwa2lkIia6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM3MDE4NjgsImhdCI6MTY3N
TkYNTg2OCwianRpIjoiOTd1Yzk3NTQtYzAzMS00OTNkLTkxZjUtOTkxZjMwOWE1OWIxIiwiXNzIjoiAHRO
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTIyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWVudHYxIiwick2Vzc21vb19zdGF0ZSI6IjVmZDg3ZWR1LWFjZDQtNGE2ZC04MmRmLTli
MmM5YWEyNDMwYyIsInNjb3BlIjoic21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjM0NiJ9.ku
ysB-RLNEgsuFBYPGkHOKU_ITk9-EozllJtbADtNdE"
}
}

```

2. initConsent

The screenshot shows a Postman collection interface. The title bar reads "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 11. Intermediary API's OTP / 4. InitConsent". The main area displays a POST request to "{{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}". The "Headers" tab is selected, showing the following configuration:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> Content-Type	application/json		
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32		
<input checked="" type="checkbox"/> PSU-ID	smartpg		
Key	Value	Description	

```

1
2     "allAccounts": [
3         {
4             "iban": "R093RZBR0000069999999999",
5             "currency": "RON"
6         }
7     ],
8     "codeBank": "OTP",
9     "periodOfValidity": 90,
10    "psuEmail": "aaaa@itsmartsystems.eu",
11    "psuIntermediaId": "324567",
12    "redirectURL": "https://google.com",

```

Response:

```

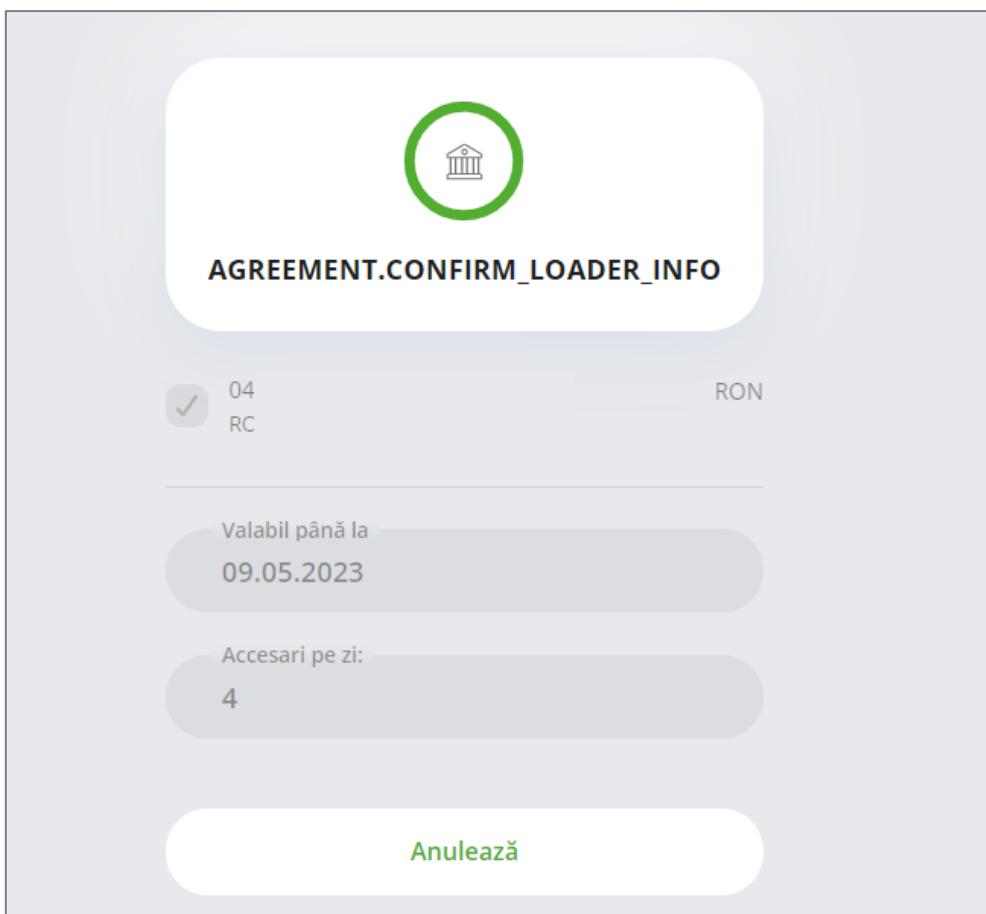
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://api.sandboxaccounts.smartfintech.eu/gateway/core/rest/api/confirmation?lang=en&state=QU1TJk9UUCZhYTA5NjNmMy1hNzAxLTQ4MjAtMmVlNC1mNGFjMTg3NjljM2UmaHR0cHM6Ly9zbWFydHBnLWh1Yi5wcm9kLmNvbS2zbWFydGZpbnR1Y2g="
      }
    }
  }
}

```

2.3 initCosent in SANDBOX – in SANDBOX environment OTP does not redirect into front-end

2.4 initConsent in PROD – in PROD environment OTP redirects into front-end. The example below shows how the PSU will interact with bank's internet banking application.





After completing fields "Utilizator:" and "Parola:", and pressing "Autentificare", thus authorising the consent, PSU or developer will be redirected to the established page.

3 GetConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 11. Intermediary API's OTP / 5. GetConsent

GET {{API_BASE_URI}}/gateway/core/rest/api/consent/OTP/{{internalConsentId}}

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies
		Headers					
		6 hidden					

KEY VALUE DESCRIPTION

PSU-IP-Address 192.168.1.103

Content-Type application/json

Key Value Description

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "id": "12345678901234567890123456789012",
    "date": "2023-05-09T10:00:00Z",
    "validUntil": "2023-05-09T10:00:00Z"
  }
}
```

```

    "access": {
        "accounts": [],
        "balances": [
            {
                "iban": "RO77OTPV116000XXXXXXR001"
            },
            {
                "iban": "RO34OTPV116000XXXXXXEU01"
            }
        ],
        "transactions": [
            {
                "iban": "RO77OTPV116000XXXXXXR001"
            },
            {
                "iban": "RO34OTPV116000XXXXXXEU01"
            }
        ],
        "availableAccounts": null
    },
    "consentStatus": "valid",
    "frequencyPerDay": null,
    "lastActionDate": null,
    "validUntil": "2021-01-30",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
}
}

```

4 GetConsentStatus

The screenshot shows a Postman request configuration for the 'GetConsentStatus' endpoint. The URL is `{{API_BASE_URI}}/gateway/core/rest/api/consent/OTP/{{internalConsentId}}/status`. The method is set to `GET`. The `Headers` tab is active, displaying the following header entries:

KEY	VALUE	DESCRIPTION	... Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32		<input type="button"/> Bulk Edit <input type="button"/> Presets
Key	Value	Description	

Response:

```

{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "valid"
    }
}

```

5 GetAccounts

The screenshot shows the Postman interface with the following details:

- Request Method:** GET
- URL:** {{API_BASE_URI}}/gateway/core/rest/api/accounts/OTP/{{internalConsentId}}
- Headers:**
 - PSU-IP-Address: 192.12.123.32
 - Key: Value
- Buttons:** Save, Send, Cookies

Response:

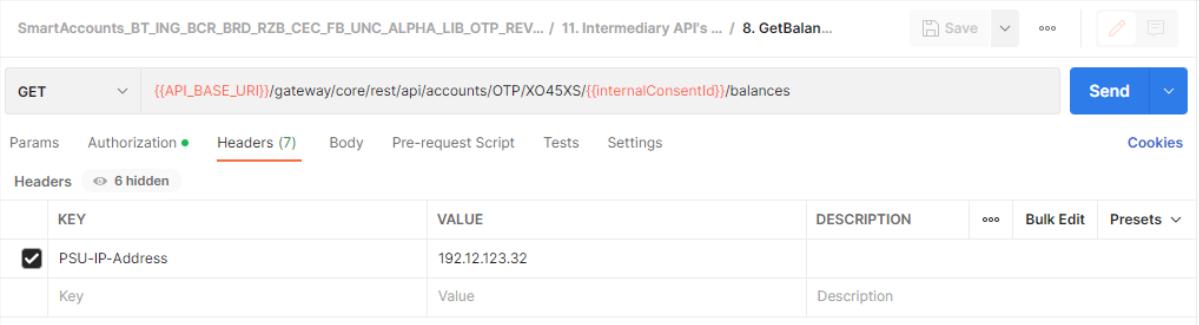
```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "XO45XS",
      "iban": "RO77OTPV116000XXXXXXR001",
      "currency": "RON",
      "name": "77-RO01",
      "product": null,
      "cashAccountType": "Cont curent",
      "status": null,
      "bic": null,
      "accountOfficerDescription": null,
      "openingDate": null,
      "customerSector": null,
      "customerType": null,
      "accountUsage": null,
      "workingBalance": null,
      "onlineActualBalance": null,
      "balances": [
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 42665.74
          },
          "balanceType": "interimBooked",
          "referenceDate": null,
          "creditLimitIncluded": null,
          "creditDebitIndicator": null
        },
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 42638.84
          }
        }
      ]
    }
  ]
}
```

```

        "balanceType": "interimAvailable",
        "referenceDate": null,
        "creditLimitIncluded": null,
        "creditDebitIndicator": null
    }
],
"sortCodeAccountNumber": null
},
{
    "resourceId": "83XISG",
    "iban": "RO09OTPV300000XXXXXXHU01",
    "currency": "HUF",
    "name": "09-HU01",
    "product": null,
    "cashAccountType": "Cont curent",
    "status": null,
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
        {
            "balanceAmount": {
                "currency": "HUF",
                "amount": 0.0
            },
            "balanceType": "interimBooked",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        },
        {
            "balanceAmount": {
                "currency": "HUF",
                "amount": 0.0
            },
            "balanceType": "interimAvailable",
            "referenceDate": null,
            "creditLimitIncluded": null,
            "creditDebitIndicator": null
        }
    ],
    "sortCodeAccountNumber": null
}
]
}

```

6 GetBalances



The screenshot shows the Postman interface with a GET request to `({API_BASE_URI})/gateway/core/rest/api/accounts/OTP/XO45XS/{internalConsentId}/balances`. The 'Headers' tab is active, displaying a table with one row: 'PSU-IP-Address' set to '192.12.123.32'. Other tabs like 'Params', 'Authorization', 'Body', etc., are visible but not selected.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO77OTPV116000XXXXXXRO01"  
        },  
        "balances": [  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 4551.39  
                },  
                "balanceType": "interimBooked",  
                "referenceDate": null,  
                "creditLimitIncluded": null,  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": null,  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            },  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 4551.39  
                },  
                "balanceType": "interimAvailable",  
                "referenceDate": null,  
                "creditLimitIncluded": null,  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": null,  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            }  
        ]  
    }  
}
```

7 GetTransactions

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_R... / 11. Intermediary API's... / 9. GetTransact...

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/OTP/XO45XS/{{internalConsentId}}/transactions?dateFrom=2022-08-01&dateTo=2022-08-20

Send

Params • Authorization • Headers (7) Body Pre-request Script Tests • Settings Cookies

Headers 6 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_R... / 11. Intermediary API's... / 9. GetTransact...

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/OTP/XO45XS/{{internalConsentId}}/transactions?dateFrom=2022-08-01&dateTo=2022-08-20

Send

Params • Authorization • Headers (7) Body Pre-request Script Tests • Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> dateFrom	2022-08-01			
<input checked="" type="checkbox"/> dateTo	2022-08-20			
<input checked="" type="checkbox"/> bookingStatus	booked			
Key	Value	Description		

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_R... / 11. Intermediary API's... / 9. GetTransact...

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/OTP/XO45XS/{{internalConsentId}}/transactions?dateFrom=2022-08-01&dateTo=2022-08-20

Send

Params • Authorization • Headers (7) Body Pre-request Script Tests • Settings Cookies

```

1 const responseJson = pm.response.json();
2 let next = responseJson.payload.transactions.links;
3 if (next != null) {
4   postman.setEnvironmentVariable("next_page", next.next.href);
5 }
6 else {
7   postman.setEnvironmentVariable("next_page", null);
8 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {},
    "transactions": {
      "booked": [
        {
          "transactionId": "120963140",
          "entryReference": null,
          "endToEndId": null,
          "bookingDate": "2017-12-18",

```

```

        "valueDate": "2017-12-18",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "RON",
            "amount": 40.0
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": null,
        "creditorAccount": {
            "iban": null
        },
        "debtorName": "internal",
        "debtorAccount": {
            "iban": ""
        },
        "remittanceInformationUnstructured": "Depunere numerar: dep num
//447428373/MOLNAR ROBERT MOLNAR ROBERT",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    },
{
        "transactionId": "144007637",
        "entryReference": null,
        "endToEndId": null,
        "bookingDate": "2018-05-22",
        "valueDate": "2018-05-22",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "RON",
            "amount": 1235.0
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": null,
        "creditorAccount": {
            "iban": null
        },
        "debtorName": "internal",
        "debtorAccount": {
            "iban": ""
        },
        "remittanceInformationUnstructured": "Incasare OP-
22313534/STRATOLYTICS SRL : SALAR 2018 Aprilie",
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
}

```

```

        ],
        "pending": [],
        "links": null
    },
    "balances": []
}
}

```

8 GetAccountDetails

The screenshot shows a Postman collection named "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1". The current request is "11. Intermediary API's OTP / 10. GetAccountDetails". The method is set to "GET" and the URL is `((API_BASE_URI))/gateway/core/rest/api/accounts/OTP/XO45XS/({internalConsentId})/details`. The "Headers" tab is active, displaying the following header:

KEY	VALUE	DESCRIPTION	... Bulk Edit Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32		
Key	Value	Description	

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "resourceId": "XO45XS",
        "iban": "RO770TPV116000XXXXXXR001",
        "currency": "RON",
        "name": "CurrentRO1",
        "product": null,
        "cashAccountType": null,
        "status": null,
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}
```

9 DeleteConsent

The screenshot shows a Postman request configuration for deleting a consent. The method is set to **DELETE** and the URL is `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/OTP/{{internalConsentId}}`. The **Headers** tab is selected, containing the following entries:

KEY	VALUE	DESCRIPTION
PSU-IP-Address	192.12.123.32	
Key	Value	Description

Response:

```
{  
  "status": 200,  
  "messageStatus": "Success",  
  "payload": {  
    "message": "Consent has been successfully revoked"  
  }  
}
```

10 RefreshToken

The screenshot shows a Postman request configuration for refreshing a token. The method is set to **POST** and the URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The **Headers** tab is selected, containing the following entries:

KEY	VALUE	DESCRIPTION
internalConsentId	{{internalConsentId}}	
access_token	{{access_token}}	
Key	Value	Description

The screenshot shows a Postman request configuration for refreshing a token. The method is set to **POST** and the URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The **Headers** tab is selected, containing the following entries:

KEY	VALUE	DESCRIPTION
refresh_token	{{refresh_token}}	
client_id	{{client_id-partner}}	
Key	Value	Description

Below the headers, there is a section for **Body** type selection, with **form-data** selected.

```

POST {{MTLS_BASE_URI}}/gateway/authenticate/rest/api/refreshToken
Params Authorization Headers (9) Body • Pre-request Script Tests • Settings Cookies
1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116346,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSlldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZZJIn0.eyJleHAiOje2NzU5MjgxMTUsImhd
CI6MTY3NTkyNzgxNSwianRpIjoijZjZhZjgxZDAtYzcwNi00Njd1LTk4YjYtMzh1ZjKzTUXy2Y5Iiwi
xXnZIjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cy
sImF1ZCI6ImFjY291bnQiLCJzdWIiOiI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLC
J0eXAiOiJCZWfYzxIiLCJhenAiOiJ0ZN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiNWZkODdlZGutY
WNkNC00YTZkLTgyZGYtOWIyYzlhYTI0MzbjIiwiYWNyIjoiMSIisImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYw5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUiOiJzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTE2
MzQ2IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2MzQ2IiwiY2xpZW50SWQiOj0ZN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xNzYuMTg4IiwiZWhxfdmVyaWZpZWQiOmZhbHN1C
JwcmVmZXJyZWRfdXN1cm5hbWUiOjzzXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc2LjE4OCJ9.AiceYT1uwvBRNNyBuyls1QiJ8S6zfqHgSd70dDPGcVaD0ZoWmMyYrXk
pUivY46WS2eoGrXVymaWDI18K-if3fpss6CDUQ-v6AarHYcBYZ4Wo2oNX2ZAcIe-
LORrximeEx15cWoY2cDSJ5I1kAIwlFN0f07vBbU5-MnGmdOpgFN61HKQXuhMw11_C88oTY-1j_m-
xjex3sLCIterEOFbP7qSCFn2sNANubXRXksdHQqmIZ-
oD3Lq4z1dQBqvXC0X1MFedn6B9PTDCQQGKXRbyrRogt2B3PYXa6UACDV0VmbIGo0lt_FyqhN6PkYjA8V497
CZ4EVNYa5Yqb_KGNcg",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSlldUIiwia2lkIiA6ICl30WNhY
2RmMi1kMjk3LTQ0YWItyjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOje2ODM3MDE4NjgsImhdCI6MTY3N
TkyNzgxNSwianRpIjojY2EyYj1jMTItNWYzNS00OTNhLTkzNmUtNjU1MzIyNTJiNjY0Iiwi
xXnZIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZX1jbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOiI2NmE2ZTiYMC11OWE0LTQ2M2ItYWFMZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOjPZmZsaW51IiwiY
XpWIjoidGVzdGNsaWVudHYxIiwiC2Vzc2lvb19zdGF0ZSI6IjVmZDg3ZWRllWFjZDQtNGE2ZC04MmRmLtlI
Mm5YWEyNDMwYyIsInNjb3BlIjoiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltOjExNjMONiJ9.1T
sRnELZpypXh5-9XtxmQxCqWs-YvD9Ggd-ihhAhxwU"
  }
}

```

In case the "accessToken" expires, the response will have the following structure:

```

{
  "status": 401,

```

```

    "messageStatus": "Unauthorized",
    "moreDetails": {
        "text": "Invalid access token"
    }
}

```

12. Garanti BBVA – Flow

1. Authenticate

The screenshot shows the Postman interface for an 'Authenticate' request. The method is POST, and the URL is {{MTLS_BASE_URL}}/gateway/authenticate/rest/api/token. The 'Body' tab is selected, showing a form-data entry for 'client_id' with the value {{client_id-partner}}. The 'Tests' tab contains a script:

```

1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 116428,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU5NDc1MzUsImIhd
CI6MTY3NTk0NzIzNSwianRpIjoiYjQwNzQ2MGUtOWMxNC00NWZmLWIzNzgtYjAwZDk3NjU3YWM0IiwiAxNz
IjoiiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1oIjgwODAvYXV0aC9yZWFSbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOiJCZWFyZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYmU0Y2E1ZmEtZ
jhmNy00NWU4LWIxMDYtZDcyMmI4MzhmN2QwIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFSbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbij
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJ
JdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCisIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnNlbRfaWRfY2xhaW06MTE2
NDI4IiwiC21hcnRhY2NvdW50c19jb25zZW50X21kX2NsYWltIjoiMTE2NDI4IiwiY2xpZW50SWQiOj0ZXN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xODMuMTQyIiwiZW1haWxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRFdXN1cm5hbWUiOjZzXJ2aN1lWfjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QRKc

```

```

mVzcyI6IjEwLjAuMTgzLjE0MiJ9.H_2oZTOCEECKA6Bd1wbt1RjrJ8jqYgc1ej0lAovcrWOLiR7q8wiHjgs
sj3KgVGOHRaA3wHlpAy2gY2n1c61xM5mufjZGYXXSMWcPyZ7b3sym5ip4zHdoQiWVxoWKF1Ycrp7TOwHKwH
Pw3A8O2TGB5d3hneqq64j9NkQCln6JG5zEEbHejwzKnHwq8vu56FkWzdoK198-s4CjchF27VPcj-mD1OX-
ULDXeYUBGI7BqjYU6mMjEtG3Vq_YoJdu1_uBqgx1v-
w8H6XHf16bp4EnbJMeozOcRFO4Lk6aGu021MA6xZYy6GPY8bMxrOxXKjWAYF8YP_1-XNV4mr0wk8ZG5A",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIia6IC13OWNhY
2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOiE2ODM3MjMyMzUsImIhdCI6MTY3N
Tk0NzIzNiwanRpIjoiYTc1YjdkNmItZDIxMC00ZTR1LWJ1MWQtODY2M2MxMDhiYWZhIiwickXNzIjoiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFBsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTiYMC11OWE0LTQ2M2ItYWFmZi1mOTc4MTU2NDc3ZDMiLCJ0eXAiOiJPZmZsaW51IiwiY
XpwIjoidGVzdGNsaWvdHYxIiwick2Vzc21vb19zdGF0ZSI6ImJ1NGNhNWZhLWY4ZjctNDV1OC1iMTA2LWQ3
MjJiODM4ZjdkMCIsInNjb3B1Ijoic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjExNjQyOCJ9.CG
Nx2VQTE1XrFYv5zFBM-RhmeQrq6aqUo-JB670iv1I"
}
}

```

2. initConsent

The screenshot shows a Postman request configuration for the 'initConsent' endpoint. The method is POST, and the URL is {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}. The 'Headers' tab is selected, showing two headers: Content-Type (application/json) and PSU-IP-Address (192.12.110.32). Other tabs like Params, Authorization, Body, and Tests are visible but not selected.

The screenshot shows a Postman request configuration for the 'initConsent' endpoint. The method is POST, and the URL is {{API_BASE_URI}}/gateway/core/rest/api/initConsent/{{internalConsentId}}. The 'Body' tab is selected, showing a JSON payload:

```

1 {
2     "codeBank": "GR", 
3     "periodOfValid": 90, 
4     "psuEmail": "aaa@itsmartsystems.eu", 
5     "psuIntermediarId": "324567", 
6     "redirectURL": "https://google.com", 
7     "TCaccepted": true
8 }

```

Other tabs like Params, Authorization, Headers, and Tests are visible but not selected.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://api.sca.oauth.com/auth/realms/master/protocol/openid-connect/token"
      }
    }
  }
}
```

```

        "href": "https://api.sandboxaccounts.smartfintech.eu/gateway/core/re
est/api/confirmation?response_type=code&client_id=jf9uzXRwWOnBkUc48Ax5&scope=openid
&redirect_uri=https://www.TPPurl.com&id=092bb59b-968a-43c8-91d3-
47e2c6bdbcf&type=c&state=eyJlbmMiOjJBMTI4Q0JDLUhTMjU2IiwiYWxnIjoizGlyIiwiY3R5Ijois
ldUIn0..QLzvozGoH53Ue7Y17ANn9g.HOD_-
BzR7tQIpPTgGk898Qn8s1E7xhZs0icuQKPsZHJ0we2GTUo82J1x4fVHBoUMJ6Ccx9gay1mgS2G5-
q9Wh6heKqFKUhwoLTh1v1RB41Z8Rz3jIdrHSGlqRfHvNU3_Yg4jP2Eo6xZsErE5v14ecx9Bn2I4X_5wKSno
_badZKiauFJQ2aXViIYoah1Y7rdtL91_HYyD-
ra1g3I9ncUy3pmCr5o97ntRrTagA2ZdPF5EbaqiuzLNz2oOBQZBM4NUF4E1omNGEv1Vj45o0iNDs_O6nKx4
eSuEwXxSaXCrsU1sryLPZP5U_mlo0c9EewbSZvOXGpMkopnHAvpHVru-w22x-
xNSRxFukbuwbCcukMbMTQIX0GWofKxj9NoEw7pIB9W10Ix72rEMzuHhJ8T6B6AGF5_5yETlhLcftSzoUOy
qYZN1Kkw5eDnmTty7h-
vfwvViV6PktfASG5WuHNntLgTFknOKeIvIndTmsH5MagZSz1or2Rw8xWz25G6soEBvjY0VGmCmb5spADXX
esNRVYX-41eMYNdyCd3-QH_RnnIWHX902nYnwJgDvfj0zq70ApFO-
812bo34QpfwqawzaGOfQ7woy5DUuBvgDs8D39XLqQpdjm-
oSztfHpmBtGzcYNZIwtwG8uiC4BCpTmjMkMWzXnrKrHpWUV8_pvu5g7GQk_kQp13Jdj9W613-
LIwtQ5ImYxKKjkOXlGf1YQyJ-AfmM40NdxnX_HxaThWpD-YHIAY-
12GZ3L0mNpIHChtCr8DqGKYJfb3kK1p3EtELO_EM2gNb6h7INTRP1OYKxq9YB-
XdV1ZGeJTs9ZKzCKsrOXk4YI-wS061Gh9cT1WoR9dZUQZKEI5866YTp68Z3EJkKSTysFxnHO3-
dDU8v19r2pGx52FjBq8o52ZEtF_6rg8i8t66VaTDy6b7c9QiBR4kvJSbThzlbULvnWmKi36mg8EmychaRi-
USw9b_NFB3Y4VaGJqENFUS3ZTOwMhRULzAzSwk51aBmBtYRlhrRMuiZ-
9CEtf8ktCYIV9DQenhb0UuPnSdo2lw38IUxfOs9bGK6jc3_mh6f5wUxz.xgq7-aHW139czVNzw5cyQQ"
    }
}
}
}
}

```

2.1 initCosentin SANDBOX – in SANDBOX environment GRT does not redirect into bank's front-end

2.1 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application

Name and surname	LUCIAN
Last login date	16/02/2023 14:18
Login method	Login via Mobile Notification
Mobile Device	Samsung · 16/02/2023 14:18
Session information	TACTICAL 16/02/2023 14:19:32 Google Chrome
To approve the mobile notification 39 seconds left.	
SEND NOTIFICATION AGAIN	

After approving mobile notification, PSU will be redirected to the established page.

3. GetConsent

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 13. Intermediary API's GRT / 5. GetConsent

GET https://{{API_BASE_URI}}/gateway/core/rest/api/consent/GRT/{{internalConsentId}}

Params Authorization Body Pre-request Script Tests Settings Cookies

Headers (8) [6 hidden](#)

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
<input checked="" type="checkbox"/> Content-Type	application/json				
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [],
      "balances": [],
      "transactions": [],
      "availableAccounts": "allAccounts"
    },
    "consentStatus": "valid",
    "frequencyPerDay": 3,
    "lastActionDate": null,
    "validUntil": "2019-10-10",
    "recurringIndicator": true,
    "combinedServiceIndicator": null,
    "creationDate": null
  }
}
```

4. GetConsentStatus

The screenshot shows a Postman request configuration. The method is GET, the URL is `({API_BASE_URI})/gateway/core/rest/api/consent/GRT/{internalConsentId}/status`, and the status bar indicates "Sending...". The "Headers" tab is selected, showing two entries: "PSU-IP-Address" with value "192.12.110.32" and "Key" with value "Value". Other tabs like Params, Authorization, Body, etc., are visible but not selected.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "valid"
  }
}
```

5. GetAccounts

The screenshot shows a Postman request configuration. The method is GET, the URL is `({API_BASE_URI})/gateway/core/rest/api/accounts/GRT/{internalConsentId}`, and the status bar indicates "Send". The "Headers" tab is selected, showing two entries: "PSU-IP-Address" with value "192.12.123.32" and "Key" with value "Value". Other tabs like Params, Authorization, Body, etc., are visible but not selected.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": [
    {
      "resourceId": "Uk8yM1VHQkkwMDAwMDEyMDEyNzI4Uk9O",
      "iban": "RO22UGBI0000012012728RON",
      "currency": "RON",
      "name": "RON Account",
      "product": null,
      "cashAccountType": "CACC",
      "status": "enabled",
      "bic": null,
      "accountOfficerDescription": null,
      "openingDate": null,
      "customerSector": null,
      "customerType": null,
      "accountUsage": null,
      "customerName": null,
      "customerAddress": null,
      "customerPhone": null,
      "customerEmail": null,
      "customerFax": null,
      "customerMobile": null,
      "customerTaxId": null,
      "customerTaxNumber": null,
      "customerTaxRegistration": null,
      "customerTaxAddress": null,
      "customerTaxPhone": null,
      "customerTaxEmail": null,
      "customerTaxFax": null,
      "customerTaxMobile": null,
      "customerTaxSector": null,
      "customerTaxType": null,
      "customerTaxUsage": null
    }
  ]
}
```

```

        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 0.26
                },
                "balanceType": "openingBooked",
                "referenceDate": "2019-02-23",
                "creditLimitIncluded": "false",
                "creditDebitIndicator": null
            }
        ],
        "sortCodeAccountNumber": null
    },
    {
        "resourceId": "Uk80NFVHQkkwMDAwMDEyMDM0MzEyUk90",
        "iban": "RO44UGBI0000012034312RON",
        "currency": "RON",
        "name": "RON Account",
        "product": null,
        "cashAccountType": "CACC",
        "status": "enabled",
        "bic": null,
        "accountOfficerDescription": null,
        "openingDate": null,
        "customerSector": null,
        "customerType": null,
        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 3.0
                },
                "balanceType": "openingBooked",
                "referenceDate": "2017-12-22",
                "creditLimitIncluded": "false",
                "creditDebitIndicator": null
            }
        ],
        "sortCodeAccountNumber": null
    }
]
}

```

6. GetBalances

The screenshot shows the Postman application interface. The URL in the header bar is `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 13. Intermediary API's GRT / 8. GetBalances`. The request method is `GET` and the endpoint is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/GRT/Uk8yMIVHQkkwMDAwMDEyMDEyNzI4Uk90/({{internalConsentId}})/balances`. The `Headers` tab is selected, showing two entries: `PSU-IP-Address` with value `192.12.123.32` and `Key` with value `Value`. Other tabs include `Params`, `Authorization`, `Body`, `Pre-request Script`, `Tests`, and `Settings`. A `Send` button is at the top right.

Response:

```
{  
    "status": 200,  
    "messageStatus": "Success",  
    "payload": {  
        "account": {  
            "iban": "RO22UGBI0000012012728RON"  
        },  
        "balances": [  
            {  
                "balanceAmount": {  
                    "currency": "RON",  
                    "amount": 0.26  
                },  
                "balanceType": "openingBooked",  
                "referenceDate": "2019-02-23",  
                "creditLimitIncluded": "false",  
                "creditDebitIndicator": null,  
                "lastChangeDateTime": "2019-02-23-04.37.26.177236",  
                "lastCommittedTransaction": null,  
                "transactionStatus": null  
            }  
        ]  
    }  
}
```

7. GetTransactions

The screenshot shows the Postman application interface. The URL in the header bar is `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 13. Intermediary API's GRT / 9. GetTransactio...`. The request method is `GET` and the endpoint is `({{API_BASE_URI}})/gateway/core/rest/api/accounts/GRT/Uk8yMIVHQkkwMDAwMDEyMDEyNzI4Uk90/({{internalConsentId}})/transactions?dateFrom=20`. The `Headers` tab is selected, showing two entries: `PSU-IP-Address` with value `192.12.123.32` and `Key` with value `Value`. Other tabs include `Params`, `Authorization`, `Body`, `Pre-request Script`, `Tests`, and `Settings`. A `Send` button is at the top right.

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 13. Intermediary API's G... / 9. GetTransactio...

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/GRT/Uk8yMIVHQkkwMDAwMDEyMDEyNzI4Uk9O/({internalConsentId})/transactions?dateFrom=2023-01-01&dateTo=2023-02-09&bookingStatus=booked

Params • Authorization • Headers (7) Body Pre-request Script Tests • Settings Cookies

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> dateFrom	2023-01-01		...	
<input checked="" type="checkbox"/> dateTo	2023-02-09		...	
<input checked="" type="checkbox"/> bookingStatus	booked		...	
Key	Value	Description	...	

SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v... / 13. Intermediary API's G... / 9. GetTransactio...

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/GRT/Uk8yMIVHQkkwMDAwMDEyMDEyNzI4Uk9O/({internalConsentId})/transactions?dateFrom=2023-01-01&dateTo=2023-02-09&bookingStatus=booked

Params • Authorization • Headers (7) Body Pre-request Script Tests • Settings Cookies

```

1 const responseJson = pm.response.json();
2 let next = responseJson.payload.transactions.links;
3 if (next != null) {
4   postman.setEnvironmentVariable("next_page", next.next.href);
5 }
6 else {
7   postman.setEnvironmentVariable("next_page", null);
8 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO22UGBI0000012012728RON"
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "20190126083235988627",
          "entryReference": "2019-01-26-08.32.35.983153",
          "endToEndId": null,
          "bookingDate": "2019-01-26",
          "valueDate": "2019-01-28",
          "executionDateTime": null,
          "transactionAmount": {
            "currency": "RON",
            "amount": 1.0
          },
          "transactionType": null,
          "exchangeRate": [],
          "creditorName": "Creditor Name",
          "creditorAccount": {
            "iban": "RO22UGBI0000012012728RON"
          }
        }
      ]
    }
  }
}
```

```

        },
        "debtorName": "Debtor Name",
        "debtorAccount": {
            "iban": "RO22UGBI0000012012728RON"
        },
        "remittanceInformationUnstructured": null,
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "smartRemittance": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    },
    {
        "transactionId": "20190223043726177236",
        "entryReference": "2019-02-23-04.37.26.173240",
        "endToEndId": null,
        "bookingDate": "2019-02-23",
        "valueDate": "2019-02-25",
        "executionDateTime": null,
        "transactionAmount": {
            "currency": "RON",
            "amount": 1.0
        },
        "transactionType": null,
        "exchangeRate": [],
        "creditorName": "Creditor Name",
        "creditorAccount": {
            "iban": "RO22UGBI0000012012728RON"
        },
        "debtorName": "Debtor Name",
        "debtorAccount": {
            "iban": "RO22UGBI0000012012728RON"
        },
        "remittanceInformationUnstructured": null,
        "proprietaryBankTransactionCode": null,
        "additionalInformation": null,
        "smartRemittance": null,
        "transactionStatus": null,
        "creditDebitIndicator": null
    }
],
"pending": [],
"links": null
},
"balances": [
{
    "balanceAmount": {
        "currency": "RON",
        "amount": 0.26
    },
    "balanceType": "openingBooked",

```

```

        "referenceDate": "2019-02-23",
        "creditLimitIncluded": "false",
        "creditDebitIndicator": null,
        "lastChangeDateTime": "2019-02-23-04.37.26.177236",
        "lastCommittedTransaction": "2019-02-23-04.37.26.173240",
        "transactionStatus": null
    }
]
}
}

```

8. GetTransactionsDetails

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "transactionId": "20190126083235988627",
    "entryReference": "2019-01-26-08.32.35.983153",
    "endToEndId": null,
    "bookingDate": "2019-01-26",
    "valueDate": "2019-09-23",
    "transactionAmount": {
      "currency": "RON",
      "amount": 1.0
    },
    "transactionType": null,
    "creditorName": "Creditor Name",
    "creditorAccount": {
      "iban": "RO22UGBI0000012012728RON"
    },
    "debtorName": "Debtor NAME",
    "debtorAccount": {
      "iban": "RO22UGBI0000012012728RON"
    },
    "remittanceInformationUnstructured": null,
    "executionDateTime": null
  }
}
```

9. GetAccountDetails

The screenshot shows the Postman interface with a GET request to `/accounts/{internalConsentId}/details`. The Headers tab is active, displaying the following header:

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "resourceId": "Uk8yM1VHQkkwMDAwMDEyMDEyNzI4Uk9O",
    "iban": "RO22UGBI0000012012728RON",
    "currency": "RON",
    "name": "Main Account",
    "product": null,
    "cashAccountType": "CACC",
    "status": "enabled",
    "bic": null,
    "accountOfficerDescription": null,
    "openingDate": null,
    "customerSector": null,
    "customerType": null,
    "accountUsage": null,
    "workingBalance": null,
    "onlineActualBalance": null,
    "balances": [
      {
        "balanceAmount": {
          "currency": "RON",
          "amount": 0.26
        },
        "balanceType": "openingBooked",
        "referenceDate": "2019-02-23",
        "creditLimitIncluded": "false",
        "creditDebitIndicator": null
      }
    ],
    "sortCodeAccountNumber": null
  }
}
```

10. DeleteConsent

The screenshot shows a Postman request configuration for a DELETE operation. The URL is `((API_BASE_URI))/gateway/core/rest/api/deleteConsent/GRT/{{internalConsentId}}`. The Headers tab is selected, showing the following header entries:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32			
Key	Value	Description		

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "message": "Consent has been successfully revoked"
  }
}
```

11. RefreshToken

The screenshot shows a Postman request configuration for a POST operation. The URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The Headers tab is selected, showing the following header entries:

KEY	VALUE	DESCRIPTION	Bulk Edit	Presets
<input checked="" type="checkbox"/> internalConsentId	{{internalConsentId}}			
<input checked="" type="checkbox"/> access_token	{{access_token}}			
Key	Value	Description		

The screenshot shows a Postman request configuration for a POST operation. The URL is `((MTLS_BASE_URI))/gateway/authenticate/rest/api/refreshToken`. The Body tab is selected, showing the following body entries:

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> refresh_token	{{refresh_token}}		
<input checked="" type="checkbox"/> client_id	{{client_id-partner}}		
Key	Value	Description	

Below the table, there are options for the body type: none, form-data (selected), x-www-form-urlencoded, raw, binary, and GraphQL.

```

POST {{MTLS_BASE_URL}}/gateway/authenticate/rest/api/refreshToken
Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies
1 var jsonData = JSON.parse(responseBody);
2 postman.setEnvironmentVariable("access_token", jsonData.payload.access_token);
3 postman.setEnvironmentVariable("refresh_token", jsonData.payload.refresh_token);
4 postman.setEnvironmentVariable("internalConsentId", jsonData.payload.internalConsentId);

```

Response:

```

{
  "status": 200,
  "messageStatus": "SUCCESS",
  "payload": {
    "internalConsentId": 116428,
    "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC14alhJWS
1YbUhFQ0I0NkZBckhZSzU0QndGYXF1Mi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOjE2NzU5NTYyMzksImhd
CI6MTY3NTk1NTkzOSwianRpIjoiMjdkZDQxOWQtYzEyOC00MjdiLWI3NWEtYmYwOWZjZjM4NjMzIiwiaXNz
IjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWiiOii2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLC
J0eXAiOjJCZWfYXZXiILCJhenAiOj0ZXN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiYmU0Y2E1ZmEtZ
jhmNy00NWU4LWIxMDYtZDcyMmI4MzhmN2QwIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKjd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiolsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdG1vbiJ
dfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2x1cyI6WyJ1bWFfcHJvdGVjdG1vbiJ
JdfSwiYWNjb3VudCI6eyJyb2x1cyI6WyJtyW5hZ2UtYWNjb3VudCisIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidm1ldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnnlbnRfaWRfY2xhaW06MTE2
NDI4Iiwigc21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYwltIjoiMTE2NDI4IiwiY2xpZW50SWQiOj0ZXN
0Y2xpZW50djEiLCJjbG11bnR1b3N0IjoiMTAuMC4xODMuMTQyIiwiZWhxfdmVyaWZpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOjzZXJ2aN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTgzLjE0MiJ9.Kv9fIipc7EVNzk7Cp00Rx0zxzdJszt36Uct-
zCDaqzDyeAA7exptffEAW7QvqAPvN5EE4MqI5YdtU-
RHf50ovoAP4x8IjRPA1V_GV6ONE_187PgGH1FzJCYU_yXBSRP8h21_BIZp_WG8GifjXxcTJS2OP2CXazt7c
-F1N4ZcUuX4M-DEQ9-wvkVQZ7su-
S5Yba0A5NO1ksgc8Qoi7mMTr7QHeWOCI2Hzx_yDuh07gEEwm1y1ie_swdPXGKm4C0MBSKND18JncncJzyc3
q7KsV77Jr4Tygp_soRiXDjIdElqYQ957o9DVUgAIm1GuB3IXF_nD45aSv6cLX6JjdspoQA",
    "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IC13OWNhY
2RmMi1kMjk3LTQ0YWIYjY5Ni05NtC0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODM3MjMyMzYsImhdCI6MTY3N
Tk1NTkzOSwianRpIjoiMGViYTkyZGYtYze1MS00ZWQ0LTg4MjAtMmQ2NDIwMmYwNWJmIiwiaXNzIjoiHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9haylhchAtc2VydmljZTo4MDgwL2F1dGgvcvMvhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTiyMC11OWE0LTQ2M2ItYWFMZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY
XpWIjoiGVzdGNsaWVudHYxIiwigc2Vzc21vb19zdGF0ZSI6ImJ1NGNhNWZhLWY4ZjctNDV1OC1iMTA2LWQ3
MjJiODM4ZjdkMCIsInNjb3BlIjoiic21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYwltOjExNjQyOCJ9.hu
UMBkOhPtcSqhubrHblcJYH4Yg1c8_dhiE23B8TiNs"
  }
}

```

In case the "accessToken" expires, the response will have the following structure:

```

{
  "status": 401,
  "messageStatus": "Unauthorized",

```

```

    "moreDetails": {
        "text": "Invalid access token"
    }
}

```

13. Libra – Flow

1. Authenticate

The screenshot shows a Postman interface with a POST request to `{{MTLS_BASE_URL}}/gateway/authenticate/rest/api/token`. The 'Body' tab is selected, containing a key-value pair: `client_id: {{client_id-partner}}`. The 'Headers' tab shows `Content-Type: application/x-www-form-urlencoded`.

Response:

```
{
    "status": 200,
    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 118352,
        "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS1YbUhFQ0I0NkZBckhZSzU0QndGYXFMi1NUUF4SEJXYmlxb2ZZJIn0.eyJleHAiOjE2ODA2Nzc2OTQsImhdCI6MTY4MDY3NzM5MywianRpIjoiOWhZWIyOTktMzU0Ni00YzAyLTg4ZDktMGNjYWN1MmEwOTRmIiwiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6ImFjY291bnQiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJCZWfYZXIiLCJhenAiOiJ0ZXN0Y2xpZW50djEiLCJzZXNzaWuX3N0YXR1IjoiMTlkMWQ3Y2EtNTMzOS00NWQwLWFmZDITYjQzMTBmZmYyNWZkIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJdLCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZv9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbiJdfSwicmVzb3VY2VfYWNjZXNzIjp7InRlc3RjbG11bnR2MSI6eyJyb2xlcI6WyJ1bWFfcHJvdGVjdGlvbiJdfSwiYWNjb3VudCI6eyJyb2xlcI6WyJtYW5hZ2UtYWNjb3VudCIsIm1hbmFnZS1hY2NvdW50LWxpbtzIiwidmlldy1wcm9maWx1I119fSwic2NvcGUIoIjzbWFydGFjY291bnRzX2NvbnN1bnRfaWRFy2xhaW06MTE4MzUyIiwiC21hcnRhY2NvdW50c19jb25zzW50X21kX2NsYWltIjoiMTE4MzUyIiwiY2xpZW50SWQiOjJ0ZXN0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4xNzguMTcyIiwiZWhfdmVyaWZpZWQiOmZhbHN1LCJwcmVmZXJyZWRfdXN1cm5hbWUiOjZZXJ2aWN1LWFjY291bnQtGvzdGnsaWVudHYxIiwiY2xpZW50QRkcMvzcyI6IjEwLjAuMTc4LjE3MiJ9.MkCnWjd0dmogIJWLze9yL18Ej8jU5cxwBNy9XN1XW0bBqpvAwaoY0AerrnOq69_1p907VYmx1fRiDV-9bCNaguoeHGbp0nky52pVM1bCCKRBG1ctLy2OX2kRufol8_puSjqEfYkdPovNZe-PCN8RT2tt6jb7j1tZRSAobtxQm73kKUBQ_C3MfxKNruyhleuT9W_wWNaEETp6-c2d0zSHzjZCNXVTdi6B3EEExsojC63gPmcWyEn1dmtNNsvTVBuonL0qGGWyFPLtRMnhOlicHd1pNyXKytoN61sb078M179yUknI-EJ9dfW1Cc8JmXwN3E8WpRkL23G_aSEbGE1Qw",
        "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS2RmMi1kMjk3LTQ0YWItYjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOjE2ODg0NTMzOTMsImhdCI6MTY4MDY3NzM5NCwianRpIjoiNWJkNjIwODEtYjZhYy00YWQxLTkxMmYtZTBiYzgxMgy4ZTI1IiwiiaXNzIjoiHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhY2NvdW50cyIsImF1ZCI6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLCJzdWIiOjI2NmE2ZTiyMC11OWE0LTQ2M2ItYWFmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOjJPZmZsaW51IiwiY"
    }
}
```

```

XpwIjoidGVzdGNsaWVudHYxIiwic2Vzc2lvb19zdGF0ZSI6IjE5ZDFkN2NhLTUzMzktNDVkMC1hZmQyLWI0
MzEwZmZmMjVmZCIsInNjb3BlIjoic2lhcncRhY2NvdW50c19jb25zzW50X21kX2NsYWltOjExODM1MiJ9._w
cB94x8cWuMgqNwPqjZ-2ECAZ7ITK--aFMpedQHfCc"
}
}

```

2. initConsent

Key	Value	Description	...	Bulk Edit	Presets
Content-Type	application/json				
PSU-IP-Address	192.12.110.32				
PSU-ID-Type	ALL				
Key	Value	Description			

```

1 {
2   "codeBank": "LIB",
3   "periodOfValid": 90,
4   "psuEmail": "aaaagitsmartsystems.eu",
5   "psuIntermediarId": "324567",
6   "redirectURL": "https://google.com",
7   "TCaccepted": true
8 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "consentStatus": "RECEIVED",
    "consentId": null,
    "_links": {
      "scaOAuth": {
        "href": "https://api.sandboxaccounts.smartfintech.eu/gateway/core/rest/api/authorization?internalConsentId=118352&bankRedirectUri=https://test.librabank.ro:9443/oauth2/authorize?state%3DeyJlbmMiOiJBMTI4Q0JDLUhTMjU2IiwiYXwnIjoiZGlyIiwiY3R5IjoiSldUIIn0..ohuTyMtcKoi4Iwbdp9jQLg.v4Oz_GNQEsXcpDNQN1g0yvvvn7ZvNe wQuRhR1TI1VtzlnNv7Exb4YzWqrAtrBXMVuB71NfrIMxqX5xBhcz3lc4rlmeYdDlPE6U2o- CmnzPxka5c1T-MIrEm69BLWC9iptZco08fwFP-u9QxCpleXjs7-vqQYBF0tAXAFrixqQWFzRidbv0zic8- s1upj62vYgAbdr1tQw8CQhKIs1wVr2W0IuKRr0TMX-AVZ-5locwShNyKQKtHfnSM2_3L- 2SlWxn6vm9VOygDRsRC4Vq9OMTIJ8onPETGwubUdcJ5_sS4jx6Wb_geGwrlwwShxncrPCuhAxk0R4nzz7N- t02Ka4liQGw59oCGBtKnNqpa6OJ_wMxpRekjggYwYNk1X3TbdwUYJYFKz9LqjgopkVMTvs_nuErbKP5vhjJ XRxDZDAL-yAENVtd0Z427vwYJ5iZLTBBJ7VnoGzuxIponMqemnUhq- kL5kxCgutNtlWB10ptVgx92SJN1pe4AwkrwlwWfioZWb8ZEbNLOHPeDXvc9tcuH74aX3ZE56Zc0gjc3nGr1 aZazyWidhZuftVYSbDok2NWY9N97T4X73PjRnzqXk613JnZDPkbBy2JpAmutCkrbzrJVf-pGw2Yf48a-
}
```

```

304zFAdeCeOFkEpwyMg9SrxXhWefIKBCfIfsPnMMAejg3hY7HRbE8ns2foYfCtPdYBztjARrGwwPTA0WRNB
6X2eqwRuBcYpwypMf3QZVfMWnCYPN5jhk78uu_HbvjTKKb9g8jbb2g95rw2FsRFnMEE5EPACKhGC7xO2DdG
zw2iPNqlvg-
hBUU259GkQ7M2HyqgyzD4_STg1Qnze9DzP47vpHZfHsP_4yqM_E9ySdEb1N8ZLpwMxg03DA5nbciQBfaSHQ
jVayvn-p3znqmI6NHsgeFN7fR4P6R3tcocSIsDG67Jpo6x2shVrg9CaIngOo46S49-
1TQzpV2ED2rUY7uiSWpqfI5BiwdsjwVkbzntmZ3aqsxICHSWOTHGHSG_rMzsJFTDxO4RcxzOsVjv-
nIrZsCLxaRwTI2KZ0P1fDVG024.Dx29U4g0DWr12dFWSmF0kQ%26response_type%3Dcode%26client_i
d%3Dn4IWqJHYT0dOqGuzckXyVyX7440a%26redirect_uri%3Dhttps://api.sandboxaccounts.smart
fintech.eu/gateway/core/rest/api/confirmation%26scope%3DAIS:5c9aef20-80b8-4053-
a2db-b22fd70393c6"
}
}
}
}

```

2.1 initCosentin SANDBOX – the example below shows how the developer will interact with bank's internet banking application

The image consists of two vertically stacked screenshots of a web-based banking application.

Screenshot 1: Initial Login Screen

- The title bar says "Conectare la Libra Internet Bank".
- The account number field contains "BGE1659277000".
- The password field contains ".....".
- Buttons at the bottom are "Renunta" (cancel) and a red "Login" button with a lock icon.

Screenshot 2: SMS Authentication Screen

- The title bar says "Autentificare prin SMS".
- A message states "Vei primi codul de logare prin SMS la numarul 004072*****15".
- The code input field contains "123".
- Buttons at the bottom are "Renunta" (cancel), a red "Valideaza" button with a lock icon, and a "Retrimite codul" link.

LIBRA INTERNET BANK

Smartaccounts solicita acces la conturile tale.

Selecteaza conturile pentru care oferi acces:

Selectati toate

<input checked="" type="checkbox"/> RO20BREL Cont online business	978.43 RON
Permisii acces informatii cont: IBAN Sold Tranzactii	
<input checked="" type="checkbox"/> RO51 Cont Online Cont Online	1.000001E7 RON
Permisii acces informatii cont: IBAN Sold Tranzactii	

Renunta **Continua**

LIBRA INTERNET BANK

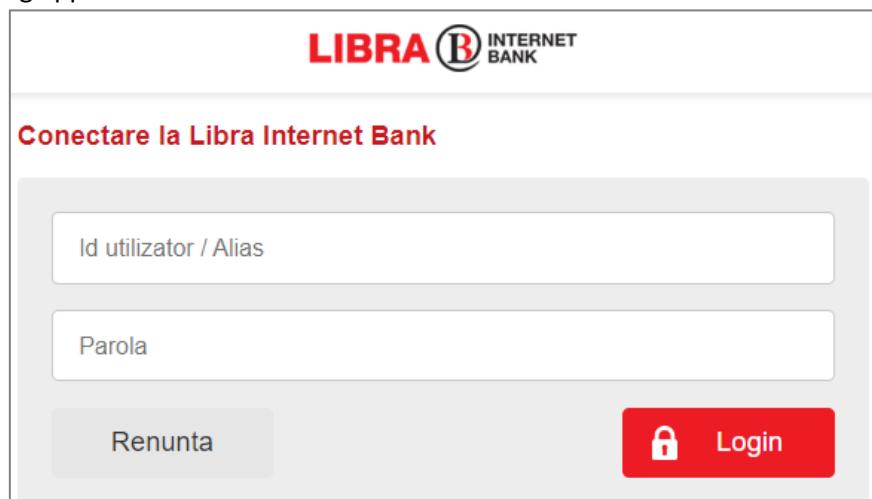
Ati oferit acces la:

RO20BREL RON
Permisii acces informatii cont: <input checked="" type="checkbox"/> IBAN <input checked="" type="checkbox"/> Sold <input checked="" type="checkbox"/> Tranzactii
RO51BREL RON
Permisii acces informatii cont: <input checked="" type="checkbox"/> IBAN <input checked="" type="checkbox"/> Sold <input checked="" type="checkbox"/> Tranzactii

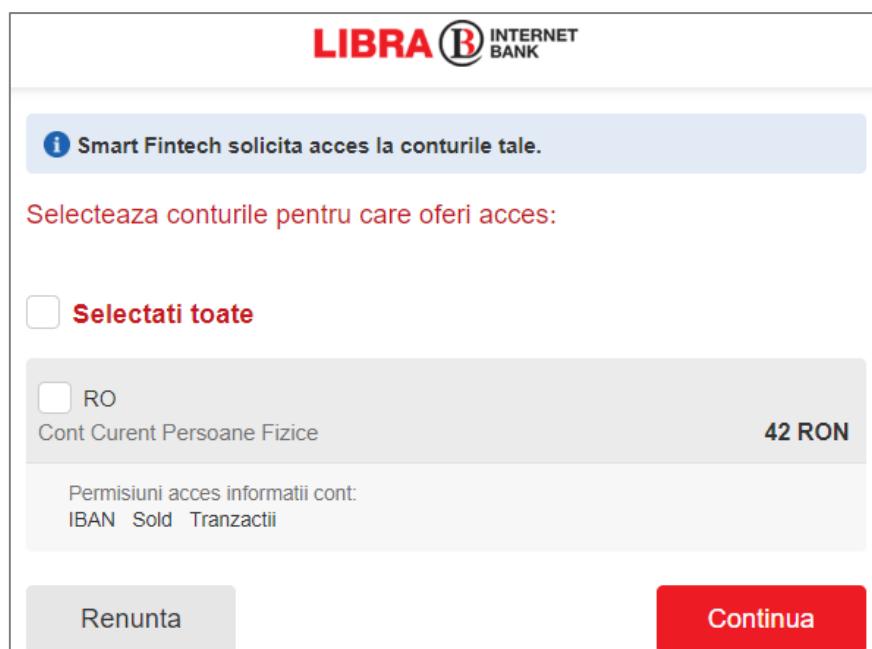
Continua

After selecting accounts and pressing "Continua", thus authorising the consent for selected account, developer will be redirected to the established page.

2.2 initConsent in PROD – the example below shows how the PSU will interact with bank's internet banking application



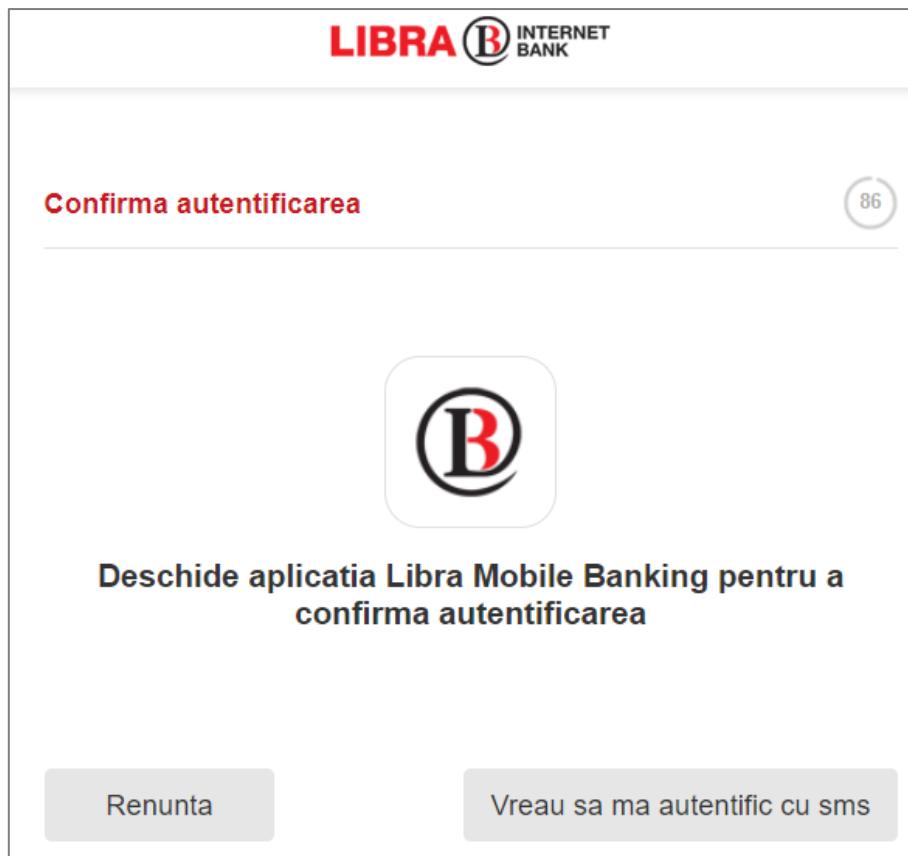
The screenshot shows the login page of LIBRA Internet Bank. At the top is the bank's logo, "LIBRA INTERNET BANK". Below it is the heading "Conectare la Libra Internet Bank". There are two input fields: "Id utilizator / Alias" and "Parola". Below the fields are two buttons: "Renunta" (Cancel) on the left and a red "Login" button on the right, which contains a lock icon.



This screenshot shows a consent request from "Smart Fintech". A message at the top says "Smart Fintech solicita acces la conturile tale." Below it, a heading reads "Selecteaza conturile pentru care oferi acces:". There is a checkbox labeled "Selectati toate". Below this, a specific account is listed: "RO Cont Curent Persoane Fizice" with a price of "42 RON". Underneath the account details, it says "Permisii acces informatii cont: IBAN Sold Tranzactii". At the bottom are "Renunta" and "Continua" buttons.



This screenshot shows the confirmation of the access grant. It displays the message "Ati oferit acces la:" followed by the account information "RO3 RON 100". Below this, it lists the selected permissions: "IBAN", "Sold", and "Tranzactii", each preceded by a checked checkbox. At the bottom is a single "Continua" button.



3. GET Consent

GET {{API_BASE_URI}}/gateway/core/rest/api/consent/LIB/{{internalConsentId}}

Send

Params	Authorization	Headers (8)	Body	Pre-request Script	Tests	Settings	Cookies
		Headers					
Headers	PSU-IP-Address	192.12.110.32					
	Content-Type	application/json					
	Key	Value	Description	...	Bulk Edit	Presets	

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "access": {
      "accounts": [
        {
          "iban": "RO00BREL000000000000000000000000",
          "currency": "RON"
        }
      ],
      "balances": [
        {
          "iban": "RO00BREL000000000000000000000000",
          "balance": 12345678901234567890
        }
      ]
    }
  }
}
```

```

        "currency": "RON"
    }
],
"transactions": [
{
    "iban": "RO00BREL0000000000000000",
    "currency": "RON"
}
],
"availableAccounts": null
},
"consentStatus": "valid",
"frequencyPerDay": 4,
"lastActionDate": "2023-04-05",
"validUntil": "2023-07-03",
"recurringIndicator": true,
"combinedServiceIndicator": false,
"creationDate": null
}
}
}

```

4. GET ConsentStatus

GET [\({{API_BASE_URI}}/gateway/core/rest/api/consent/LIB/{{internalConsentId}}/status\)]({{API_BASE_URI}}/gateway/core/rest/api/consent/LIB/{{internalConsentId}}/status)

Params	Authorization	Headers (7)	Body	Pre-request Script	Tests	Settings	Cookies
Headers <small>(6 hidden)</small>							
		Key	Value	Description	<small>...</small>	Bulk Edit	Presets
		<input checked="" type="checkbox"/> PSU-IP-Address	192.12.110.32				
		Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "consentStatus": "VALID"
    }
}
```

5. GET Accounts

GET [\({{API_BASE_URI}}/gateway/core/rest/api/accounts/LIB/{{internalConsentId}}\)]({{API_BASE_URI}}/gateway/core/rest/api/accounts/LIB/{{internalConsentId}})

Params	Authorization	Headers (7)	Body	Pre-request Script	Tests	Settings	Cookies
Headers <small>(6 hidden)</small>							
		Key	Value	Description	<small>...</small>	Bulk Edit	Presets
		<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
		Key	Value	Description			

Response:

```
        "status": 200,  
        "messageStatus": "Success",  
        "payload": [  
            {  
                "resourceId": "1000000000000000",  
                "iban": "RO00BREL00000000000000000000",  
                "currency": "RON",  
                "name": "ABC S.R.L.",  
                "product": "ABC S.R.L.",  
                "cashAccountType": null,  
                "status": null,  
                "bic": null,  
                "accountOfficerDescription": null,  
                "openingDate": null,  
                "customerSector": null,  
                "customerType": null,  
                "accountUsage": null,  
                "workingBalance": null,  
                "onlineActualBalance": null,  
                "balances": [],  
                "sortCodeAccountNumber": null  
            }  
        ]  
    }  
}
```

6. GET Balances

GET {{API_BASE_URL}}/gateway/core/rest/api/accounts/LIB/1000000000000/{{internalConsentId}}/balances

Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Headers (6 hidden)

Key	Value	Description	ooo	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
    "status": 200,
    "messageStatus": "Success",
    "payload": {
        "account": {
            "iban": "RO00BREL00000000000000000000"
        },
        "balances": [
            {
                "balanceAmount": {
                    "currency": "RON",
                    "amount": 978.43
                }
            }
        ]
    }
}
```

```

        "balanceType": "expected",
        "referenceDate": "2023-04-05",
        "creditLimitIncluded": null,
        "creditDebitIndicator": null,
        "lastChangeDateTime": "2023-04-05 09:59:12",
        "lastCommittedTransaction": null,
        "transactionStatus": null
    }
]
}
}

```

7. GET Transactions

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/LIB/100000000000/{{internalConsentId}}/transactions?dateFrom=2023-02-01&dateTo=2023-04-05

Params • **Authorization** • **Headers (7)** **Body** **Pre-request Script** **Tests** • **Settings** **Cookies**

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> dateFrom	2023-02-01			
<input checked="" type="checkbox"/> dateTo	2023-04-05			
Key	Value	Description		

GET {{API_BASE_URI}}/gateway/core/rest/api/accounts/LIB/100000000000/{{internalConsentId}}/transactions?dateFrom=2023-02-01&dateTo=2023-04-05

Params • **Authorization** • **Headers (7)** **Body** **Pre-request Script** **Tests** • **Settings** **Cookies**

Headers (6 hidden)

Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "account": {
      "iban": "RO00BREL0000000000000000"
    },
    "transactions": {
      "booked": [
        {
          "transactionId": "FT000000ZPVZ",
          "entryReference": null,
          "endToEndId": null,
          "bookingDate": "2023-01-27",
          "valueDate": "2023-01-27",
          "executionDateTime": null,
          "transactionAmount": {
            "currency": "RON",
            "amount": 1000
          }
        }
      ]
    }
  }
}
```

```
        "amount": -4.0
    },
    "transactionType": null,
    "exchangeRate": [],
    "creditorName": null,
    "creditorAccount": {},
    "debtorName": null,
    "debtorAccount": {},
    "remittanceInformationUnstructured": "Speze transfer",
    "proprietaryBankTransactionCode": null,
    "merchantDetails": null,
    "additionalInformation": null,
    "smartRemittance": null,
    "transactionStatus": null,
    "creditDebitIndicator": null
}
],
"pending": [],
"links": null
},
"balances": []
}
}
```

8. GET Accounts Details



The screenshot shows the Postman interface for a GET request. The URL is {{API_BASE_URL}}/gateway/core/rest/api/accounts/LIB/100000000000/{{internalConsentId}}/details. The Headers tab is selected, showing a key-value pair: PSU-IP-Address: 192.12.123.32. Other tabs include Params, Authorization, Body, Pre-request Script, Tests, Settings, and Cookies. A 'Send' button is at the top right.

Key	Value	Description	...	Bulk Edit	Presets
PSU-IP-Address	192.12.123.32				
Key	Value	Description			

Response:

```

        "accountUsage": null,
        "workingBalance": null,
        "onlineActualBalance": null,
        "balances": [],
        "sortCodeAccountNumber": null
    }
}

```

9. Delete Consent

The screenshot shows a Postman request configuration for a DELETE operation. The URL is `({{API_BASE_URI}})/gateway/core/rest/api/deleteConsent/LIB/{{internalConsentId}}`. The 'Headers' tab is selected, showing two entries: 'PSU-IP-Address' with value '192.12.123.32' and an empty 'Key' entry with an empty 'Value'. Other tabs include Params, Authorization, Body, Pre-request Script, Tests, Settings, and Cookies.

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "message": "Consent has been successfully revoked"
  }
}
```

10. RefreshToken

The screenshot shows a Postman request configuration for a POST operation. The URL is `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/refreshToken`. The 'Headers' tab is selected, showing two entries: 'internalConsentId' with value `{{internalConsentId}}` and 'access_token' with value `{{access_token}}`. Other tabs include Params, Authorization, Body, Pre-request Script, Tests, Settings, and Cookies.

The screenshot shows a Postman request configuration for a POST operation. The URL is `({{MTLS_BASE_URI}})/gateway/authenticate/rest/api/refreshToken`. The 'Body' tab is selected, showing two entries: 'refresh_token' with value `{{refresh_token}}` and 'client_id' with value `{{client_id-partner}}`. The 'Body' dropdown menu at the top indicates the body type is 'form-data'. Other tabs include Params, Authorization, Headers, Pre-request Script, Tests, Settings, and Cookies.

Response:

```
{
  "status": 200,
```

```

    "messageStatus": "SUCCESS",
    "payload": {
        "internalConsentId": 118352,
        "accessToken": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl4alhJWS
1YbUhFQ0I0NkZBckhSzU0QndGYXFiMi1NUUF4SEJXYmlxb2ZJIn0.eyJleHAiOje2ODA2ODE0NjgsImhd
CI6MTY4MDY4MTE2OCwianRpIjoiOWI0M2Q4NWItNjUxOS00NWizLWI3MDMtODE0MTVjYWEwMjFhIiwiaXNz
IjoiaHR0cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhy2NvdW50cyI
sImF1ZCI6ImFjY291bnQiLCJzdWIiOii2NmE2ZTiYMC11LOWE0LTQ2M2ItYWFmZilmOTc4MTU2NDc3ZDMiLc
J0eXAiOijCZWfYZXiilCJhenAiOij0ZN0Y2xpZW50djEiLCJzZXNzaW9uX3N0YXR1IjoiMTlkMWQ3Y2EtN
TMzOS00NWQwLWFmZDITYjQzMTBmZmYyNWZkIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyIvKiJd
LCJyZWFsbV9hY2N1c3MiOnsicm9sZXMiOlsib2ZmbGluZV9hY2N1c3MiLCJ1bWFfYXV0aG9yaXphdGlvbij
dfSwicmVzb3VY2VfYWNjZXnIjpb7InRlc3RjbG11bnR2MSI6eyJyb2xlcYI6WyJ1bWFfcHJvdGVjdGlvbij
JdfSwiYWNjb3VudCI6eyJyb2xlcYI6WyJtYW5hZ2UtYWNjb3VudCisIm1hbmFnZS1hY2NvdW50LWxpbtzI
iwidm11dy1wcm9maWx1I119fSwic2NvcGUIoJzbWFydGFjY291bnRzX2NvbnNlbnRfaWRfY2xhaW06MTE4
MzUyIiwc21hcnRhy2NvdW50c19jb25zzW50X21kX2NsYWltIjoiMTE4MzUyIiwiY2xpZW50SWQiOij0ZN
0Y2xpZW50djEiLCJjbG11bnRIB3N0IjoiMTAuMC4xNzguMTcyIiwiZWhxfdmVyaWzpZWQiOmZhbHN1LC
JwcmVmZXJyZWRfdXN1cm5hbWUiOijzZXJ2aWN1LWFjY291bnQtdGVzdGNsaWVudHYxIiwiY2xpZW50QWRkc
mVzcyI6IjEwLjAuMTc4LjE3MiJ9.fd08B5NUse_m73S4OqlRtrwxjijvHWh4kEKjgzcawMKd1tNTDRzZ14C
8NJV0bhNY1bR_alJBcuGcMSdS943k7Z9viJSnaKV5-
NKctLqphAN_Jrbntqs7IYbUXGc_CjXCcyih8pYbGpTHiCqN5nbtMiSbxZ9sE_Dc0HdRmvzexzOyrpWXn7B1
wFF1dR2OYhgj1VsTVv_nyzjexcqeCem6bpHkpXn0EhyLRC8vdLE9d5JI0boNJHY9nx6ruTn8n1kPPhvFy08
4xWV6o6EOj7WMaNq4ns8mZoIAx_RJ2J-jqHLacNsMt6WNwpGmZ0BwFu3ypTQEWRP7LtOeeY1Y6201g",
        "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICl30WNhY
2RmMi1kMjk3LTQ0YWItyjY5Ni05NTc0ODRmYTU2ZDYifQ.eyJleHAiOje2ODg0NTMzOTQsImhdCI6MTY4M
DY4MTE2OCwianRpIjoiYWy2ODQ5ZTctYQwNS00Y2YwLWEyNWYtNTQxMDk4YmE5YTQ4IiwiiaXNzIjoiiaHR0
cDovL2tleWNsb2FrLWFwcC1zZXJ2aWN1OjgwODAvYXV0aC9yZWFsbXMvc21hcnRhy2NvdW50cyIsImF1ZCI
6Imh0dHA6Ly9rZXljbG9hay1hcHAtc2VydmljZTo4MDgwL2F1dGgvcmVhbG1zL3NtYXJ0YWNjb3VudHMiLC
JzdWIiOii2NmE2ZTiYMC11LOWE0LTQ2M2ItYWFmZilmOTc4MTU2NDc3ZDMiLCJ0eXAiOijPZmZsaW51IiwiY
XpwljoidGVzdGNsaWVudHYxIiwiC2Vzc2lvb19zdGF0ZSI6IjE5ZDFkN2NhLTUzMzktNDVkMC1hZmQyLWI0
MzEwZmZmMjVmZCIsInNjb3BlIjoiic21hcnRhy2NvdW50c19jb25zzW50X21kX2NsYWltOjExODM1MiJ9.gK
-ZcyWH2qKvd58bHnOHTra-oFN3LFPpTIDC0MX2VBw"
    }
}

```

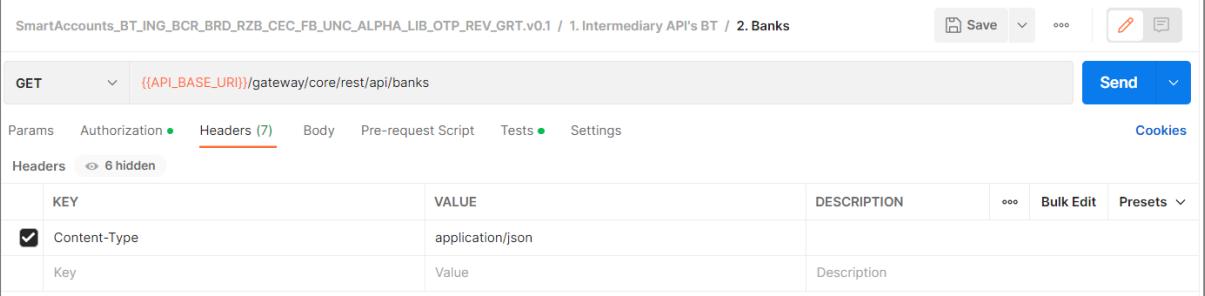
In case the "accessToken" expires, the response will have the following structure:

```
{
    "status": 401,
    "messageStatus": "Unauthorized",
    "moreDetails": {
        "text": "Invalid access token"
    }
}
```

7. Generic APIs (available in all banks' flows)

For every bank there will be a GET "Banks" and a POST "CheckEmail" calls. These APIs can be called after the authentication of the TPP – (POST) "Authenticate".

1. Banks



The screenshot shows a Postman collection named "SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1" with a selected item "2. Banks". The request method is "GET" and the URL is "({API_BASE_URI})/gateway/core/rest/api/banks". The "Headers" tab is selected, showing two entries: "Content-Type: application/json" and "Key: Value". Other tabs like "Params", "Authorization", "Body", "Pre-request Script", "Tests", and "Settings" are visible. A "Send" button is at the top right, and a "Cookies" section is at the bottom right.

Response:

```
{
    "id": 1,
    "code": "BT",
    "name": "Banca Transilvania",
    "logo": "...",
    "active": true,
    "bic": "BTRL",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    },
    "id": 2,
    "code": "ING",
    "name": "ING Bank",
    "logo": "...",
    "active": true,
    "bic": "INGB",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    },
    "id": 3,
    "code": "BRD",
    "name": "BRD Group Societe Generale",
```

```

    "logo": "...",
    "active": true,
    "bic": "BRDE",
    "requiredParameters": {
        "requiresPSUIdType": true,
        "requiresPSUId": true,
        "requiresIban": false
    }
},
{
    "id": 4,
    "code": "BCR",
    "name": "BCR Bank",
    "logo": "...",
    "active": true,
    "bic": "RNCB",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    }
},
{
    "id": 5,
    "code": "RZB",
    "name": "Raiffeisen Bank SA",
    "logo": "...",
    "active": true,
    "bic": "RZBR",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": true,
        "requiresIban": true
    }
},
{
    "id": 6,
    "code": "CEC",
    "name": "CEC Bank SA",
    "logo": "...",
    "active": true,
    "bic": "CECE",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    }
},
{
    "id": 7,
    "code": "FB",

```

```

        "name": "First Bank SA",
        "logo": "...",
        "active": true,
        "bic": "PIRB",
        "requiredParameters": {
            "requiresPSUIdType": false,
            "requiresPSUId": false,
            "requiresIban": false
        }
    },
    {
        "id": 8,
        "code": "UNC",
        "name": "UniCredit Bank SA",
        "logo": "...",
        "active": true,
        "bic": "BACX",
        "requiredParameters": {
            "requiresPSUIdType": false,
            "requiresPSUId": false,
            "requiresIban": false
        }
    },
    {
        "id": 9,
        "code": "ALPH",
        "name": "Alpha Bank",
        "logo": "...",
        "active": true,
        "bic": "BUCU",
        "requiredParameters": {
            "requiresPSUIdType": false,
            "requiresPSUId": false,
            "requiresIban": false
        }
    },
    {
        "id": 10,
        "code": "LIB",
        "name": "Libra Bank",
        "logo": "...",
        "active": true,
        "bic": "BREL",
        "requiredParameters": {
            "requiresPSUIdType": false,
            "requiresPSUId": false,
            "requiresIban": false
        }
    },
    {
        "id": 11,

```

```

    "code": "REV",
    "name": "Revolut LTD",
    "logo": "...",
    "active": true,
    "bic": "REVO",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    }
},
{
    "id": 12,
    "code": "OTP",
    "name": "OTP Bank",
    "logo": "...",
    "active": true,
    "bic": "...",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": true,
        "requiresIban": true
    }
},
{
    "id": 13,
    "code": "GRT",
    "name": "Garanti Bank",
    "logo": "...",
    "active": true,
    "bic": "...",
    "requiredParameters": {
        "requiresPSUIdType": false,
        "requiresPSUId": false,
        "requiresIban": false
    }
}
}

```

2. CheckEmail

The screenshot shows a Postman collection named 'SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1' with a single endpoint '1. Intermediary API's BT / 3. CheckEmail'. The request is a POST to `((API_BASE_URI))/gateway/core/rest/api/checkPsuEmail`. The 'Headers' tab is selected, showing two entries: 'Content-Type: application/json' and 'Key: Value'.

KEY	VALUE	DESCRIPTION	... Preset	Bulk Edit	Presets
Content-Type	application/json				
Key	Value	Description			

The screenshot shows the Postman application interface. At the top, it displays the URL: `SmartAccounts_BT_ING_BCR_BRD_RZB_CEC_FB_UNC_ALPHA_LIB OTP_REV_GRT.v0.1 / 1. Intermediary API's BT / 3. CheckEmail`. Below this, the method is set to `POST` and the endpoint is `((API_BASE_URI))/gateway/core/rest/api/checkPsuEmail`. The `Body` tab is selected, showing the following JSON payload:

```

1 {
2   "psuEmail": "bogdan.gurita@itsmartsystems.eu"
3 }

```

Response:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "existingAccount": true,
    "message": "Sunt de acord cu Termenii si Conditiiile Smart Fintech",
    "t&c": "https://www.smartfintech.eu/terms-and-conditions-for-smart-
accounts"
  }
}
```

The "false" response should have the following structure:

```
{
  "status": 200,
  "messageStatus": "Success",
  "payload": {
    "existingAccount": false,
    "message": "Sunt de acord cu Termenii si Conditiiile Smart Fintech si cu
crearea unui cont utilizand adresa de email bogdan.gurita@itsmartsystems.eu",
    "t&c": "https://www.smartfintech.eu/terms-and-conditions-for-smart-
accounts"
  }
}
```

8. Error mapping & handling linked to a consent's status

Each bank raises a different error and with different error messages (or even responds with success) on every API when a consent is not valid.

When a consent expired, was overwritten, or deleted from bank's app (where applicable):

Error group #1: GetConsent and GetConsentStatus will all respond as following:

```
{  
    "status": 401,  
    "messageStatus": "Unauthorized",  
    "source": "Bank",  
    "moreDetails": {  
        "text": "UNAUTHORIZED" OR "EXPIRED" OR "TERMINATEDBYTPP" OR "REVOKEDEBYPSPU"  
    }  
}
```

Error group #2: GetAccounts, GetBalances, GetTransactions, GetAccountDetails and GetTransactionDetails, DeleteConsent will all respond as following:

```
{  
    "status": 401,  
    "messageStatus": "Unauthorized",  
    "source": "Bank",  
    "moreDetails": {  
        "text": "UNAUTHORIZED" OR "EXPIRED"  
    }  
}
```

When you receive errors mentioned for APIs in **Error group #2** (moreDetails.text: "UNAUTHORIZED" / "EXPIRED" / "TERMINATEDBYTPP" / "REVOKEDEBYPSPU") coupled with errors mentioned for APIs in **Error group #1** (moreDetails.text: "UNAUTHORIZED" / "EXPIRED"), then this means that a new consent must be initiated because the previous consent is not valid anymore.