

PSD2 APIs testing documentation

Contents

1	Automatic enrolment:	3
1.1	Creating the user:	4
1.2	Creating an application:	7
2	Description of PSD2-CECAPIC product features.....	11
2.1	Flow description for obtaining customer accounts data	13
2.1.1	Send request for taking over customer account access agreement: create consent. .	14
2.1.2	Customer authentication in the bank system	16
2.1.3	Customer agreement to consult account data	17
2.1.4	Application access code generation	18
2.1.5	Get access tokens to consult accounts	19
2.1.6	Get refresh_token	20
2.1.7	Get account data	21
2.1.8	Get account details	24
2.1.9	Get account balance	28
2.1.10	Get transaction list by account	31
2.1.11	Get transaction details	36
2.2	Initiate and authorize payments using PSD2 API-PaymentsC service operations	39
2.2.1	POST operation.....	41
2.2.2	The TPP application must redirect the PSU to the bank login page.....	44
2.2.3	The TPP application redirects the customer to the URL formed in point 2.2.2.....	45
2.2.4	If successfully logged in, the customer can view the initiated payment data from the TPP application.....	46
2.2.5	If the payment has been authorized by the customer, a code is sent to the redirect address of the TPP application, which can be used by the TPP application to obtain the token to access the payment status.....	48

2.2.6	The TPP application will provide the above access code in exchange for a token to access the payment data (Bearer Token).....	48
2.2.7	Payment status.....	50
2.3	Funds confirmation service - PIIS functionality	53
2.4	Initiate and authorize bulk payments using PSD2 API-PaymentsC service operations	54
2.4.1	The (PSU) customer requests the payment initiation in the TPP application.	55
2.4.2	The Bank Service checks the correctness of the data and stores the payment data ...	56
2.4.3	The TPP application redirects the customer to the authorization interface exposed by the bank	57
2.4.4	After successful authentication, the customer agrees to process the bulk payment or can reject it.....	58
2.4.5	The customer is redirected to the TPP application.	58
2.4.6	The TPP application obtains tokens to access data and bulk payment status.	59

1 Automatic enrolment:

In order to start the APIs testing, user enrollment is required. To be able to do this, follow these steps:

- Creating the user
- Creating an application



[Back](#) | [Our Products](#)

[PRODUCTS](#) ▾

[HOW IT WORKS](#)

[CONTACT US](#)

[EN](#) ▾



Our Products

[← Back to Home Page](#)

- APIs for payments
- APIs for consent
- APIs for account information
- APIs for Automation Enrollment

[How it works](#)

APIs for Payments (PISP)

Provide the CEC Bank users with money transfer services, directly from your platform or application.

[READ MORE](#)



APIs for Consent

Ensure the customer's account has the funds needed for a certain transaction.

[READ MORE](#)



In the portal we will go to the "Our Products" window and select the api: API's for Automation Enrollment.

After selection we will see the "Dynamic Users Register" api:

The screenshot shows the CEC Bank API Developer Portal. The header includes the CEC Bank logo, navigation links for PRODUCTS, HOW IT WORKS, and CONTACT US, a language selector set to EN, and a user profile icon. Below the header, a breadcrumb trail reads: Back | API Products | Dynamic Users Register - API Developer Portal | Dynamic Users Register - API Developer Portal. The main title is 'Dynamic Users Register' with a version badge '1.0.0'. A left sidebar contains a menu with 'Overview' (selected), 'POST /portal/users/register', 'POST /create/app', and 'Definitions'. The main content area, titled 'Overview', displays the following details:

- Type:** REST OAS2
- Protocol:** HTTPS
- Endpoint:** Production, Development: <https://api-test.cec.ro/cec/tpp/dynamic-users-register>

At the bottom of the overview section is a link: [Download OpenAPI Document](#) with a download icon.

1.1 Creating the user:

The first call must be selected: "POST /portal/users/register" to make the user creation call.

The data filled in the body must be retained as it will be used in the application creation call.

In the body part, the following body model must be filled in:

```
body : {  
  "username": "john.doe@yahoo.com",  
  "password": "password",  
  "first_name": "John",  
  "last_name": "Doe",  
  "organization": "JohnORG"  
}
```

The screenshot displays the CEC Bank API testing interface. The header includes the CEC Bank logo, navigation links for PRODUCTS, HOW IT WORKS, and CONTACT US, a language selector set to EN, and a user profile icon. The main content area is divided into two tabs: DETAILS and TRY IT. The TRY IT tab is active, showing a POST request configuration for the endpoint `https://api-test.cec.ro/cec/tpp/dynamic-users-register/portal/users/register`. The configuration includes a Header section with an Accept field set to `application/json` and a Content-Type field also set to `application/json`. There is a TPP-Signature-Certificate field with a Generate button and a text area containing a placeholder for a certificate. The Body section has a Body* field with a Generate button and a text area containing a JSON body model: `{ "first_name": "John", "last_name": "Doe", "organization": "JohnORG" }`. At the bottom, there are RESET and SEND buttons.

CEC Bank

PRODUCTS v HOW IT WORKS CONTACT US EN

DETAILS TRY IT

POST Production, `https://api-test.cec.ro/cec/tpp/dynamic-users-register/portal/users/register`

Parameters

Header ^

Accept

`application/json`

Content-Type

`application/json`

TPP-Signature-Certificate* **Generate**

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

Body ^

Body* **Generate**

generated

```
{  
  "first_name": "John",  
  "last_name": "Doe",  
  "organization": "JohnORG"  
}
```


RESET SEND

If the data entered are valid, the call will return a "201 created" response and you will receive a confirmation email on the email address you filled in the body with the activation link.


The link must be clicked on the webpage to complete the registration step.

1.2 Creating an application:

The second call must be selected: "POST /create/app" to make the call creating the application and getting a set of test credentials.



PRODUCTS ▾HOW IT WORKSCONTACT US

EN ▾

Back | API Products | Dynamic Users Register - API Developer Portal | Dynamic Users Register - API Developer Portal

Dynamic Users Register

1.0.0

- Overview
- POST /portal/users/register
- POST /create/app
- Definitions

POST : /create/app

DETAILSTRY IT

POST Production, <https://api-test.cec.ro/cec/tpp/dynamic-users-register/create/app>
Development:

Parameters

Header ^

Accept

application/json

Content-Type

application/json

TPP-Signature-Certificate*

Generate

Redirect-URI*

Generate



Dynamic Users Register

1.0.0

- Overview
- POST /portal/users/register
- POST /create/app
- Definitions

POST : /create/app

DETAILS

TRY IT

POST

Production, <https://api-test.cec.ro/cec/tpp/dynamic-users-register/create/app>
Development:

Parameters

Header ^

Accept

application/json

Content-Type

application/json

TPP-Signature-Certificate*

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxxx...

Redirect-URI*

Generate

<http://vulo.mz/fopna>

The following headers must be filled in in order to make the call:

-TPP-Signature-Certificate = -----BEGIN CERTIFICATE-----

xxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxE
XAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxEXA
MPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxEXAMPL
ExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxEXAMPL
ExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxxxxxxxEXAMPLExxxxx-----END
CERTIFICATE-----

- Content-Type = application/json

- Redirect-Uri = "<https://www.google.com>" (the passed link is an example and you can pass any redirect link)

In the body part, the following body model must be filled in:

```
body : {  
  "username":"john.doe@yahoo.com",  
  "password":"password",  
  "app_name":"JohnAPP",  
  "app_title":"JohnAPP",  
  "organization":"JohnORG"  
}
```

If the organization name filled in the enrollment step contains spaces ("organization":"John ORG"), when filling in the request body for the application creation call we have to replace the spaces with "-".

E.g.:

```
body : {  
  "username":"john.doe@yahoo.com",  
  "password":"password",  
  "app_name":"JohnAPP",  
  "app_title":"JohnAPP",  
  "organization":"John-ORG"  
}
```



Parameters

Header ^

Accept

application/json

Content-Type

application/json

TPP-Signature-Certificate*

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

Redirect-URI*

Generate

http://vulo.mz/fopna

Body ^

Body*

Generate

generated

```
{
  "username": "john.doe@yahoo.com",
  "password": "password",
  "app_name": "JohnAPP",
  "app_title": "JohnAPP"
}
```

RESET

SEND

Following the call we will receive a formal reply:

201 Created

```
{
  "Application_Name": " JohnAPP",
  "Application_Title": " JohnAPP",
  "client_id": "f9dTESTa5798TESTc76a837a",
  "client_secret": "b0fTEST5a70TEST2e8460TEST23b50"
}
```

With these credentials we will be able to make the calls shown below and they must be saved to have them whenever you need to use them.

2 Description of PSD2-CECAPIC product features

After logging in on the portal, registered users have access to the PSD2-CECAPIC product

CEC Bank

Back | Our Products

PRODUCTS v HOW IT WORKS CONTACT US EN

Our Products

[Back to Home Page](#)

- APIs for payments
- APIs for consent
- APIs for account information
- APIs for Automation Enrollment

How it works

APIs for Payments (PISP)

Provide the CEC Bank users with money transfer services, directly from your platform or application.

[READ MORE](#)

APIs for Consent

Ensure the customer's account has the funds needed for a certain transaction.

[READ MORE](#)

This product includes the APIs needed to develop TPP applications with AIS/PIS functionalities:

- PSD2 API-ConsentC: contains the operations necessary to obtain the customer's consent for accessing accounts opened with CEC Bank;
- PSD2 API-AccountsC: contains the operations necessary to develop AIS functionalities (data about accounts is extracted with the limitations specified in the customer consent);

- In order to access/test the above APIs, the TPP application must subscribe to use them. When calling API operations, application authentication parameters are required: Client Id and Client Secret. Also, calling APIs: PSD2 API-ConsentC, PSD2 API-AccountsC, PSD2 API-PaymentsC is allowed only if the Tpp-Signature-Certificate header parameter of each call is filled with the EIDAS certificate of the TPP of the form:

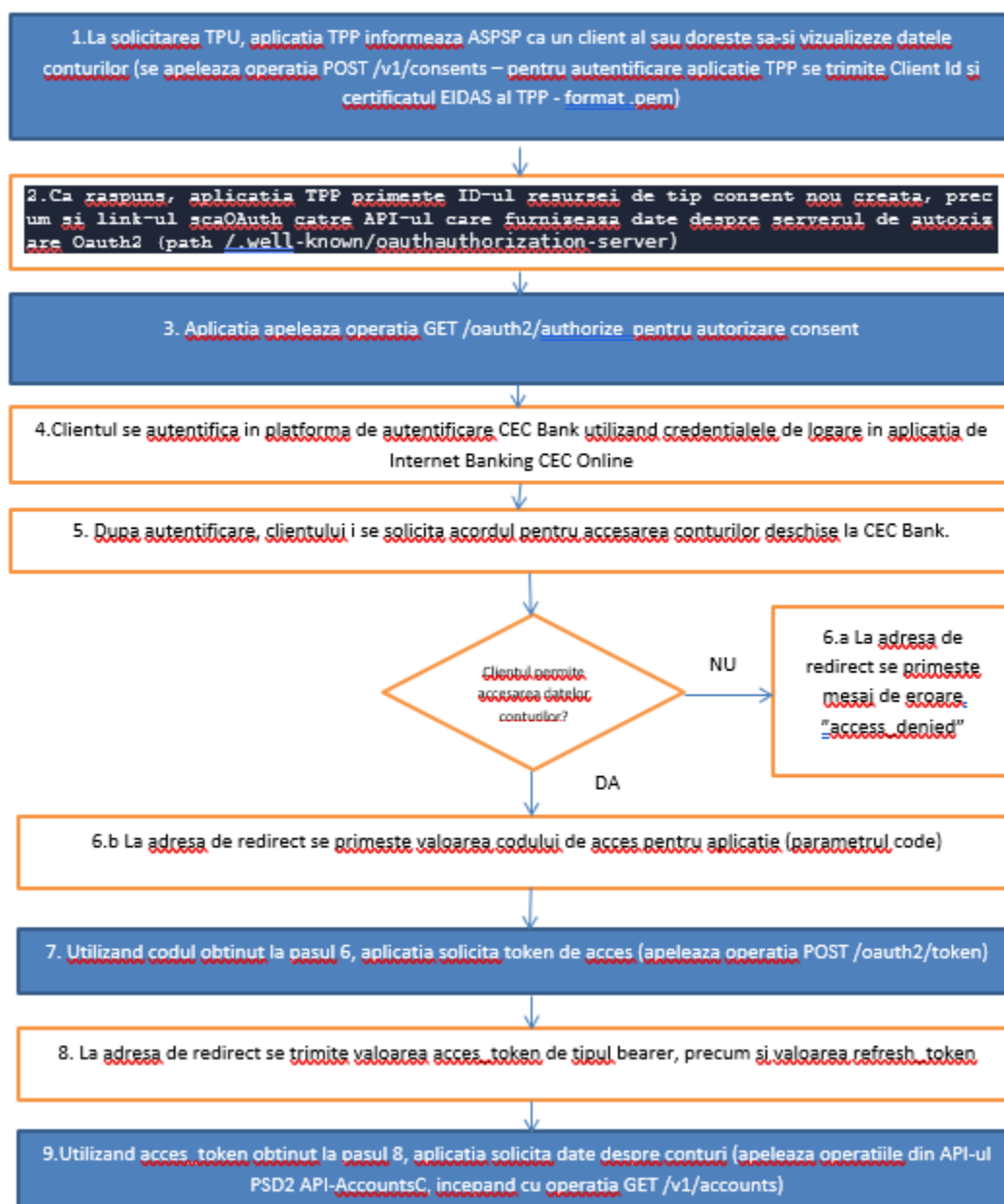
```

xxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxE
XAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXA
MPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPL
ExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxxxxxEXAMPLExxxxx-----END
CERTIFICATE-----

```

<PUBLIC>

2.1 Flow description for obtaining customer accounts data



2.1.1 Send request for taking over customer account access agreement: create consent.

Call the POST operation /v1/consents in the PSD2 API API-Consent. In the Header the following is mandatory:

- X-Request-ID
- TPP-Redirect-URI
- Tpp-Signature-Certificate.

In the "Security" section you will see the Log In button and after logging in, the subscribed product application will be automatically selected.

In Body no information should be passed on. The content of the consent will be determined by the customer after logging into the bank's system, in the consent authorization interface exposed by the bank.

The screenshot shows the CEC Bank API documentation interface. At the top, there is a navigation bar with the CEC Bank logo, links for PRODUCTS, HOW IT WORKS, and CONTACT US, and a language selector set to EN. A sidebar on the left contains a list of API endpoints, with 'POST /v1/consents' highlighted. The main content area is titled 'Create consent' and features two tabs: 'DETAILS' and 'TRY IT'. The 'DETAILS' tab is active, showing the following information:

- POST** Production, Development: <https://api-test.cec.ro/cec/tp/psd29c/v1/consents>
- Security**: Identification (dropdown), API Key (TestAPP - Crede, dropdown), and a 'Copy client ID' link. A note states: 'Only subscribed credentials are selectable.'
- Parameters**: Header (dropdown), Accept (application/json), Content-Type (application/json), and X-Request-ID (dropdown). A 'Generate' button is located at the bottom right of the parameters section.

Response

✓ 201 Created

Body

Headers

```
{
  "consentStatus": "received",
  "consentId": "c9740868-b3de-11ef-a368-0a0800410000",
  "_links": {
    "scaOAuth": {
      "href": "https://api-test.cec.ro/cec/tpp/oauth/.well-known/oauthauthorization-server"
    },
    "self": {
      "href": "https://api-test.cec.ro/cec/tpp/psd29c/v1/consents/c9740868-b3de-11ef-a368-0a0800410000"
    },
    "status": {
      "href": "https://api-test.cec.ro/cec/tpp/psd29c/v1/consents/c9740868-b3de-11ef-a368-0a0800410000/status"
    }
  }
}
```

In the response structure, the identifier of the newly created consent resource "consentId" is received, as well as the link "scaOAuth" where the metadata of the OAuth 2.0 authorization server is provided.

The TPP application must redirect the PSU to the bank authentication page, using the authorization endpoint provided by the scaOAuth link.



The screenshot shows a web browser with the address bar displaying <https://api-test.cec.ro/cec/tpp/oauth/.well-known/oauthauthorization-server>. The page content displays the JSON metadata for the OAuth 2.0 authorization server, including the issuer, authorization endpoint, token endpoint, response types supported, response modes supported, grant types supported, authorization code, ID token signing algorithm values supported, ID token encryption algorithm values supported, ID token encryption enc values supported, and the point of contact (poc).

The example of the URL formation for Redirect can be found in the "Location" attribute of the response header of the POST operation /v1/consents:

Location:

<https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/authorize?>

response_type=code

&scope=AIS:43a5a1a0-a456-11ec-b4e2c-00a0800270000

&redirect_uri=https://example.com/redirect

&client_id=28xxxxxxxxxxfe8a3

&code_challenge=<code challenge value>

&code_challenge_method=<code challenge method>

Before use, the link in the "Location" header must be filled in by the TPP application with the value of the code_challenge parameter (PKCE challenge according to cryptographic RFC 7636 - <https://tools.ietf.org/html/rfc7636> used to prevent code injection attacks, code_challenge_method= "S256")

2.1.2 Customer authentication in the bank system

The TPP application redirects the client to the authorization URL formed in the previous step, where the login window in the authentication platform opens.

The customer can log in using the username from the Internet Banking-CEC Online application and the code generated by the code generating device or eToken application.

api-test.cec.ro/cec/tpp/oauthcec/oauth2/authorize?response_type=code&scope=AIS:a2308f60-76e3-11eb-9881-0a0

CEC Bank

CEC Bank - Authorization Platform

User CEOnline
testcecap

Cod acces
.....

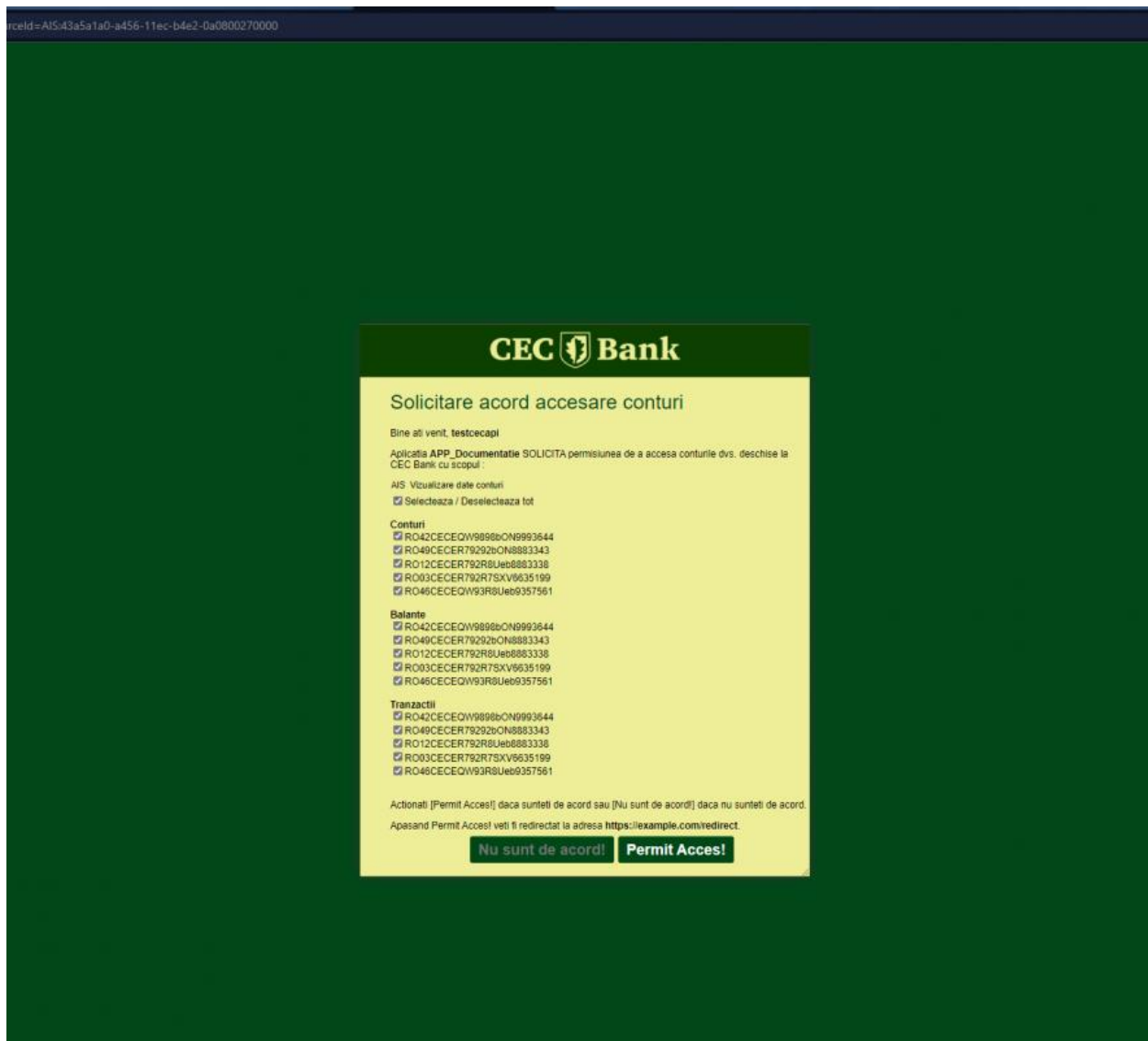
Log in

Introduceti Numele dvs de utilizator din aplicatia CEOnline si Codul de acces generat de dispozitivul de autentificare.

2.1.3 Customer agreement to consult account data

After successfully logging in and identifying the logged in customer, the agreement takeover window for accessing the accounts by the TPP application appears.

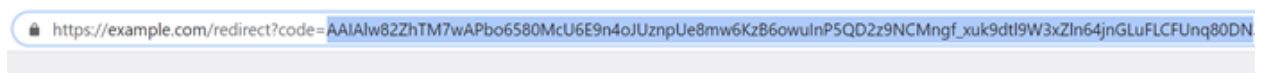
Depending on the permissions granted by the customer for each payment account in the list, the TPP application will be able to visualize the accounts' IBANs and will have access to the balance visualization, respectively to the transaction list



If you click 'I don't agree!', the redirect address of the TPP application will send "error=acces_denied"

2.1.4 Application access code generation

After the customer allows access to the accounts, the access code for the application ("code") is sent to the redirect address provided by the TPP application.



2.1.5 Get access tokens to consult accounts

The TPP application will provide the above access code in exchange for a resource access token (Bearer Token), by calling the POST method /oauth2/token from the URL indicated in the metadata "token_endpoint", with the purpose "AIS:consentId" (the same purpose contained in the access code provided in the previous step)

The request body must be sent in *x-www-form-urlencoded* format and must contain the parameters below (the values shown correspond to the in/out parameters of the authorization operation presented in the previous step):

- grant_type=authorization_code
- client_id=28xxxxxxxxxxfe8a3
- code= AAIAIw82ZhTM7wAPbo6580McU6E9n4oJUznpUe8mw6KzB6owuInP5QD2z9NCMngf_xuk9dtl9W3xZln64jnGLuFLCFUnq80DN3CaEGTxfkwE5fp2I17DuxPacZovpRCGplsj7Vfo9lqphLkdxPDVKL3j
- redirect_uri= <https://example.com/redirect>
- scope= AIS:43a5a1a0-a456-11ec-b4e2c-00a0800270000
- code_verifier=ginfeftejiscu

```
curl --request POST \
```

```
--url https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/token \  
--header 'accept: application/json' \  
--header 'content-type: application/x-www-form-urlencoded' \  
--header 'x-ibm-client-id: 28xxxxxxxxxxfe8a3\  
--data 'grant_type=authorization_code&client_id=28xxxxxxxxxxfe8a3&code=  
AAIAIw82ZhTM7wAPbo6580McU6E9n4oJUznpUe8mw6KzB6owuInP5QD2z9NCMngf_xuk9  
dtl9W3xZln64jnGLuFLCFUnq80DN3CaEGTxfkwE5fp2I17DuxPacZovpRCGplsj7Vfo9lqphLk  
dxPDVKL3j&redirect_uri= https://example.com/redirect&scope= AIS:a2308f60-76e3-11eb-  
9881-0a06001a0000&code_verifier=ginfeftejiscu'
```

200 OK

```
{
```

```
  "token_type": "bearer",
```

```

    "access_token":
    "AAIkMjg2Yjk1NmYtZDQxMS00OTM4LTljMGUtYmIODE1YWZlOGZlPNvUaB8KIeDg
PgFbjI7AANcFyG-HCnV-hiqPVlo17k5DjRoCB7PSv8--
qKI1JuneC52uXWAgU7iE1OA_NYC6JJTRYtYBi6I2DL88f4_uxormO0Bi6zBj6lld1D3rtNI9u
T0dUAL2WHk5L5VjhJFQ",

    "expires_in": 3595,

    "consented_on": 1614240924,

    "scope": "AIS:43a5a1a0-a456-11ec-b4e2c-00a0800270000",

    "refresh_token": "AAJzRQj3N2Wd-
3pb005516_i9ISl6ZL3Ut5n1w2gbh8YK5vxjb86zMQcwILT29Dq-
4ZNHvcvBSr90uoemjU9LhUgkF0-dpHPTdShGuPJOWi6gCbfaD3PSFu-
oTgKQzReCKxIOxPldguuu5M7Xquy4NudJ7LDVWRxI52JBNpxtCGYQ",

    "refresh_token_expires_in": 2682000

}

```

In the response of the POST operation /oauth2/token, the access token value required by the calls of the APIs PSD2 API-AccountsC, PSD2 API-ConsentC and the POST operation /v1/funds-confirmations in PSD2 API-PaymentsC is found. It also specifies the refresh_token value, with a validity period corresponding to the validity of the consent (max. 180 days), which can be used to generate a new access_token without PSU involvement.

2.1.6 Get refresh_token

To get a new valid access_token in exchange for the refresh_token, use the same POST /oauth2/token operation, but with grant_type=refresh_token, as in the example below.

```

curl --request POST \
--url https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/token \
--header 'accept: application/json' \
--header 'content-type: application/x-www-form-urlencoded' \
--header 'x-ibm-client-id: 28xxxxxxxxxxfe8a3\
--data 'grant_type=refresh_token&client_id=28xxxxxxxxxxfe8a3&refresh_token=
AAJzRQj3N2Wd-3pb005516_i9ISl6ZL3Ut5n1w2gbh8YK5vxjb86zMQcwILT29Dq-
4ZNHvcvBSr90uoemjU9LhUgkF0-dpHPTdShGuPJOWi6gCbfaD3PSFu-
oTgKQzReCKxIOxPldguuu5M7Xquy4NudJ7LDVWRxI52JBNpxtCGYQ

```

2.1.7 Get account data

The GET /v1/accounts operation is called from the PSD2-AccountsC API, transmitting in the header:


"x-ibm-client-id" : "the client_id generated in the application creation step of the Dynamic Enrollment api"

"x-Request-ID": "id such as: 123e4567-e89b-12d3-a456-426655440000"


"Authorization": "access the Bearer token obtained in the previous step",

"Consent-ID": "consentId value for viewing account data",

"TPP-Signature-Certificate": "EIDAS certificate of the TPP with PSP_AI role"



PRODUCTS ▾ HOW IT WORKS CONTACT US

EN 

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-AccountsC - API Developer Portal

PSD2 API-AccountsC

1.0.0

- Overview
- GET /v1/accounts
- GET /v1/d/accounts
- GET /v1/accounts/{account...}
- GET /v1/d/accounts/{accou...}
- GET /v1/accounts/{account...}
- GET /v1/d/accounts/{accou...}
- GET /v1/accounts/{account...}
- GET /v1/d/accounts/{accou...}
- GET /v1/accounts/{account...}
- GET /v1/d/accounts/{accou...}
- GET /v1/card-accounts
- GET /v1/card-accounts/{acc...}
- GET /v1/card-accounts/{acc...}
- GET /v1/card-accounts/{acc...}
- Definitions

getAccountList

DETAILS TRY IT

GET

Production, <https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts>
Development:

Security

Identification ^

API Key Copy client ID

TestAPP - Crede: ▾

Only subscribed credentials are selectable.

API Secret



Parameters

Header



Accept

application/json

X-Request-ID

Generate

2209d600-ce02-53f6-9096-09f33973748c

Consent-ID

Generate

8867797977268224

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This m...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxEXAMPLExxxxxxxxEXAMPLExxxx...

PSU-IP-Address

Generate

Response

200 OK

Body

Headers

```
{
  "accounts": [
    {
      "resourceId": "25511978",
      "iban": "RO42CECEAG0202RON0007466",
      "currency": "RON",
      "product": "Cont curent - persoane fizice - LE
I",
      "cashAccountType": "CurrentAccount",
      "name": "421642",
      "balances": [
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 26.46
          },
          "balanceType": "closingBooked",
          "referenceDate": "2024-12-06"
        }
      ],
      "_links": {
        "balances": {
          "href": "/v1/accounts/25511978/balance
s"
        },
        "transactions": {
          "href": "/v1/accounts/25511978/transac
tions"
```

2.1.8 Get account details

CEC Bank

PRODUCTS

V

HOW IT WORKS

CONTACT US

EN

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-AccountsC - API Developer Portal

PSD2 API-AccountsC

1.0.0

- Overview
- GET /v/accounts
- GET /v/d/accounts
- GET /v/accounts/{acco...
- GET /v/d/accounts/{accou...
- GET /v/accounts/{account...
- GET /v/d/accounts/{accou...
- GET /v/accounts/{account...
- GET /v/d/accounts/{accou...
- GET /v/accounts/{account...
- GET /v/d/accounts/{accou...
- GET /v/card-accounts
- GET /v/card-accounts/{acc...
- GET /v/d/card-accounts/{acc...
- GET /v/card-accounts/{acc...
- GET /v/d/card-accounts/{acc...
- Definitions

readAccountDetails

DETAILS

TRY IT

GET

Production, Development:

https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts/{account-id}

Security

Identification

API Key

TestAPP - Crede

Copy client ID

Only subscribed credentials are selectable.

API Secret

Authorization

Authorization URLs

https://192.168.2.31:443/cec/tpp/bauthcec/bauth2/authorize

Token URLs

CEC Bank

PRODUCTS

HOW IT WORKS

CONTACT US

EN

Accept

application/json

X-Request-ID

Generate

8ad42a4c-abb5-54b5-a826-a184ecec797b

Consent-ID

Generate

7097487540092928

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This m...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

PSU-IP-Address

Generate

The forwarded IP Address header field consists of the correspondi...

PSU-IP-Port

Generate

The forwarded IP Port header field consists of the corresponding





PSU-Device-ID

Generate

UUID (Universally Unique Identifier) for a device, which is used by ...

PSU-Geo-Location

Generate

The forwarded Geo Location of the corresponding http request b...

Path



account-id*

Generate

6334995949908698

Query



withBalance

true



RESET

SEND

Response

✓ 200 OK

Body

Headers

```
{
  "accounts": [
    {
      "resourceId": "25511978",
      "iban": "RO42CECEAG0202RON0007466",
      "currency": "RON",
      "product": "Cont curent - persoane fizice - LE
I",
      "cashAccountType": "CurrentAccount",
      "name": "421642",
      "balances": [
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 26.46
          },
          "balanceType": "closingBooked",
          "referenceDate": "2024-12-06"
        }
      ],
      "_links": {
        "balances": {
          "href": "/v1/accounts/25511978/balance
s"
        },
        "transactions": {
          "href": "/v1/accounts/25511978/transac
tions"
```

Call the GET /v1/accounts/{account-id} operation in the PSD2-AccountsC API as follows:

In the Header the following is transmitted:

"x-ibm-client-id" : " the client_id generated in the application creation step of the Dynamic Enrollment api "

"x-Request-ID": "id such as: 123e4567-e89b-12d3-a456-426655440000"

"Authorization": "access the Bearer token obtained in the previous step",

"Consent-ID": "consentId value for viewing account data",

"TPP-Signature-Certificate": "EIDAS certificate of the TPP with PSP_AI role"


Parameters:


"account-id": "Account id for which details are required"

Response:

```
{
  "accounts": [
    {
      "resourceId": "25511978",
      "iban": "RO42CECEQW9898bON9993644",
      "currency": "RON",
      "product": "Cont curent - persoane fizice - LEI",
      "cashAccountType": "CurrentAccount",
      "name": "421642",
      "balances": [
        {
          "balanceAmount": {
            "currency": "RON",
            "amount": 320.02
          },
          "balanceType": "closingBooked",
          "referenceDate": "2022-03-16"
        }
      ],
      "links": {
        "balances": {
          "href": "/v1/accounts/25511978/balances"
        },
        "transactions": {
          "href": "/v1/accounts/25511978/transactions"
        }
      }
    }
  ]
}
```

2.1.9 Get account balance



[PRODUCTS](#) [v](#) [HOW IT WORKS](#) [CONTACT US](#) [EN](#) 

[Back](#) | [API Products](#) | [PSD2-CEC-APIC - API Developer Portal](#) | [PSD2 API-AccountsC - API Developer Portal](#)

PSD2 API-AccountsC

1.0.0

- Overview
- GET /v/accounts
- GET /v/d/accounts
- GET /v/accounts/{account-id}
- GET /v/d/accounts/{account-id}
- GET /v/accounts/{account-id}/balances
- GET /v/d/accounts/{account-id}/balances
- GET /v/accounts/{account-id}/card-accounts
- GET /v/d/accounts/{account-id}/card-accounts
- GET /v/accounts/{account-id}/card-accounts/{card-account-id}
- GET /v/d/accounts/{account-id}/card-accounts/{card-account-id}
- Definitions

getBalances

DETAILS

TRY IT

GET

Production, <https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts/{account-id}/balances>

Developmentes

Security

Identification

API Key

TestAPP - Crede

Copy client ID

Only subscribed credentials are selectable.

API Secret

Authorization

Authorization URLs

<https://192.168.2.31:443/cec/tpp/oauthcec/oauth2/authorize>

Token URLs

<PUBLIC>



Parameters

Header ^

Accept

application/json

X-Request-ID

Generate

c85c4804-978e-522e-8f58-a76ec074ef44

Consent-ID

Generate

6420072971108352

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This mi...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

PSU-IP-Address

Generate



▼

PSU-Device-ID

Generate

UUID (Universally Unique Identifier) for a device, which is used by ...

PSU-Geo-Location

Generate

The forwarded Geo Location of the corresponding http request b...

Path ^

account-id*

Generate

5102600656365155

RESET

SEND

Response

✓ 200 OK

Body

Headers

```
{
  "account": {
    "iban": "RO42CECEAG0202RON0007466",
    "currency": "RON"
  },
  "balances": [
    {
      "balanceAmount": {
        "currency": "RON",
        "amount": "26.46"
      },
      "balanceType": "closingBooked",
      "referenceDate": "2024-12-06"
    }
  ]
}
```

Call the GET /v1/accounts/{account-id}/balances operation in the PSD2-AccountsC API as follows:

In the Header the following is transmitted:

"x-ibm-client-id" : "client_id generated by the application created on the portal"

"x-Request-ID": "id such as: 123e4567-e89b-12d3-a456-426655440000"

"Authorization": "access the Bearer token obtained in the previous step",

"Consent-ID": "consentId value for viewing account data",

"TPP-Signature-Certificate": "EIDAS certificate of the TPP with PSP_AI role"


Parameters:

"account-id": "Account id for which details are required"


Response:

```
{
  "account": {
    "iban": "RO42CECEQW9898bON9993644",
    "currency": "RON"
  },
  "balances": [
    {
      "balanceAmount": {
        "currency": "RON",
        "amount": "320.02"
      },
      "balanceType": "closingBooked",
      "referenceDate": "2022-03-16"
    }
  ]
}
```

2.1.10 Get transaction list by account



PRODUCTS ▾HOW IT WORKSCONTACT US

EN ▾

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-AccountsC - API Developer Portal

PSD2 API-AccountsC

1.0.0

- Overview
- GET /v/accounts
- GET /v/d/accounts
- GET /v/accounts/{account-id}
- GET /v/d/accounts/{account-id}
- GET /v/accounts/{account-id}/balances
- GET /v/d/accounts/{account-id}/balances
- GET /v/accounts/{account-id}/transactions
- GET /v/d/accounts/{account-id}/transactions
- GET /v/card-accounts
- GET /v/card-accounts/{account-id}
- GET /v/d/card-accounts/{account-id}
- GET /v/card-accounts/{account-id}/balances
- GET /v/d/card-accounts/{account-id}/balances
- Definitions

getTransactionList

DETAILSDetailsTRY IT

GET

Production, <https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts/{account-id}/transactions>
Development

Security

Identification

API Key

Copy client ID

TestAPP - Crede

Only subscribed credentials are selectable.

API Secret

Authorization

Authorization URLs

<https://192.168.2.31:443/cec/tpp/oauthcec/oauth2/authorize>

Token URLs



Parameters

Header ^

Accept

application/json

X-Request-ID

Generate

c85c4804-978e-522e-8f58-a76ec074ef44

Consent-ID

Generate

6420072971108352

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This m...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxEXAMPLExxxxxxxxxxEXAMPLExxxx...

PSU-IP-Address

Generate



The forwarded Geo Location of the corresponding http request b...

Path ^

account-id*

Generate

5610580470623499

Query ^

bookingStatus*

booked ▾

dateFrom

07/08/2024 📅

dateTo

07/10/2024 📅

entryReferenceFrom

Generate

jesdearfumunef

deltaList

withBalance

true ▾

Response

✓ 200 OK

Body

Headers

```
{
  "account": {
    "iban": "RO42CECEAG0202RON0007466"
  },
  "transactions": {
    "booked": [
      {
        "transactionId": "1566614557",
        "creditorName": "",
        "creditorAccount": {
          "iban": ""
        },
        "transactionAmount": {
          "currency": "RON",
          "amount": -102.59
        },
        "bookingDate": "2024-12-06",
        "valueDate": "2024-12-06",
        "remittanceInformationUnstructured": "Capitalizare dobanda-Achitare dobanda neta la scadenta din cont curent pentru contractul nr 207493400"
      },
      {
        "transactionId": "1566614556",
        "creditorName": "",
        "creditorAccount": {
          "iban": ""
        }
      }
    ]
  }
}
```

Call the GET /v1/accounts/{account-id}/transactions operation in the PSD2-AccountsC API as follows:

In the Header the following is transmitted:

"x-ibm-client-id" : "client_id generated by the application created on the portal"

"x-Request-ID": "id such as: 123e4567-e89b-12d3-a456-426655440000"

"Authorization": "access the Bearer token obtained in the previous step",

"Consent-ID": "consentId value for viewing account data",

"TPP-Signature-Certificate": "EIDAS certificate of the TPP with PSP_AI role"

Parameters:

"bookingStatus": can be: "booked", "pending", "both"

"dateFrom": "start date interval search transactions"

"dateTo": "end date interval search transactions"

"account-id": "Account id for which details are required"


Answer:

```
{
  "account": {
    "iban": "RO42CECEQW9898bON9993644"
  },
  "transactions": {
    "booked": [
      {
        "transactionId": "1174425360",
        "creditorAccount": {},
        "transactionAmount": {
          "currency": "RON",
          "amount": -19.8
        },
        "bookingDate": "2021-10-10",
        "valueDate": "2021-10-10",
        "remittanceInformationUnstructured": "Sqpytqlyzqru torqntq-
Qsxytqru torqntq nutq lq ssqtuntq tyn sont surunt puntru sontrqstul nr 894414336"
      },
      {
        "transactionId": "1174425359",
        "creditorAccount": {},
        "transactionAmount": {
          "currency": "RON",
          "amount": -2.2
        },
        "bookingDate": "2021-10-10",
        "valueDate": "2021-10-10",
        "remittanceInformationUnstructured": "Plqtq ympozyt-
Plqtq ympozyt lq ssqtuntq tyn sont surunt puntru sontrqstul nr 894414336"
      },
      {
        "transactionId": "1174425358",
        "debtorAccount": {},


```

```
"transactionAmount": {
  "currency": "RON",
  "amount": 22
},
"bookingDate": "2021-10-10",
"valueDate": "2021-10-10",
"remittanceInformationUnstructured": "Sqpytqlyzqru torqntq-
Sqpytqlyzqru torqntq rrutq lq ssqtuntq yn sont surunt puntru sontrqstul nr 894414336"
},
```

2.1.11 Get transaction details



PRODUCTS ▾HOW IT WORKSCONTACT US

EN ▾

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-AccountsC - API Developer Portal

PSD2 API-AccountsC

1.0.0

- Overview
- GET /v/accounts
- GET /v/d/accounts
- GET /v/accounts/{account-id}
- GET /v/d/accounts/{account-id}
- GET /v/accounts/{account-id}/transactions
- GET /v/d/accounts/{account-id}/transactions
- GET /v/accounts/{account-id}/card-accounts
- GET /v/d/accounts/{account-id}/card-accounts
- GET /v/accounts/{account-id}/card-accounts/{card-account-id}
- GET /v/d/accounts/{account-id}/card-accounts/{card-account-id}
- GET /v/accounts/{account-id}/card-accounts/{card-account-id}/transactions
- GET /v/d/accounts/{account-id}/card-accounts/{card-account-id}/transactions
- Definitions

getTransactionDetails

DETAILS

TRY IT

GET

Production, <https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts/{account-id}/transactions/{resourceId}>
Development, [https://api-test.cec.ro/cec/tpp/psd29c/v1/accounts/{account-id}/transactions/{resourceId}](#)

Security

Identification

API Key

TestAPP - Crede

Copy client ID

Only subscribed credentials are selectable.


API Secret

Authorization


Authorization URLs

<https://192.168.2.31:443/cec/tpp/oauthcec/oauth2/authorize>

Token URLs



PRODUCTS ▾HOW IT WORKSCONTACT US

EN ▾

Parameters

Header

Accept

application/json

X-Request-ID

Generate

c85c4804-978e-522e-8f58-a76ec074ef44

Consent-ID

Generate

6420072971108352

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This mi...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

PSU-IP-Address

Generate

The forwarded Geo Location of the corresponding http request b...

Path

^

account-id*

Generate

4903051341521048

resourceId*

Generate

6466182968246272

RESET

SEND

Response

✓ 200 OK

Body

Headers

```
{
  "transactions": {
    "booked": [
      {
        "transactionId": "1444173462",
        "creditorName": "",
        "creditorAccount": {
          "iban": ""
        },
        "transactionAmount": {
          "currency": "RON",
          "amount": -271.46
        },
        "bookingDate": "2023-10-11",
        "valueDate": "2023-10-11",
        "remittanceInformationUnstructured": "Capi-  
talizare dobanda-Achitare dobanda neta la scadenta din con-  
t curent pentru contractul nr 209304160"
      }
    ]
  },
  "account": {
    "iban": "RO42CECEAG0202RON0007466"
  }
}
```

Call the GET /v1/accounts/{account-id}/transactions/{resourceId} operation in the PSD2-AccountsC API as follows:

In the Header the following is transmitted:

"x-ibm-client-id" : "client_id generated by the application created on the portal"

"x-Request-ID": "id such as: 123e4567-e89b-12d3-a456-426655440000"

"Authorization": "access the Bearer token obtained in the previous step",

"Consent-ID": "consentId value for viewing account data",

"TPP-Signature-Certificate": "EIDAS certificate of the TPP with PSP_AI role"

Parameters:

"account-id": "Account id for which details are required"

"resourceId": "Transaction Id"

Response:

```
{
  "transactions": {
    "booked": [
      {
        "transactionId": "1174425360",
        "creditorAccount": {},
        "transactionAmount": {
          "currency": "RON",
          "amount": -19.8
        },
        "bookingDate": "2021-10-10",
        "valueDate": "2021-10-10",
        "remittanceInformationUnstructured": "Sqpytqlyzqru torqntq-
Qsxytqru torqntq nutq lq ssqtuntq tyn sont surunt puntru sontrqstul nr 894414336"
      },
    ],
  },
}
```

```

    {
      "debtorName": "",
      "debtorAccount": {
        "iban": ""
      },
      "transactionAmount": {}
    }
  ]
},
"account": {
  "iban": "RO42CECEQW9898bON9993644"
}
}

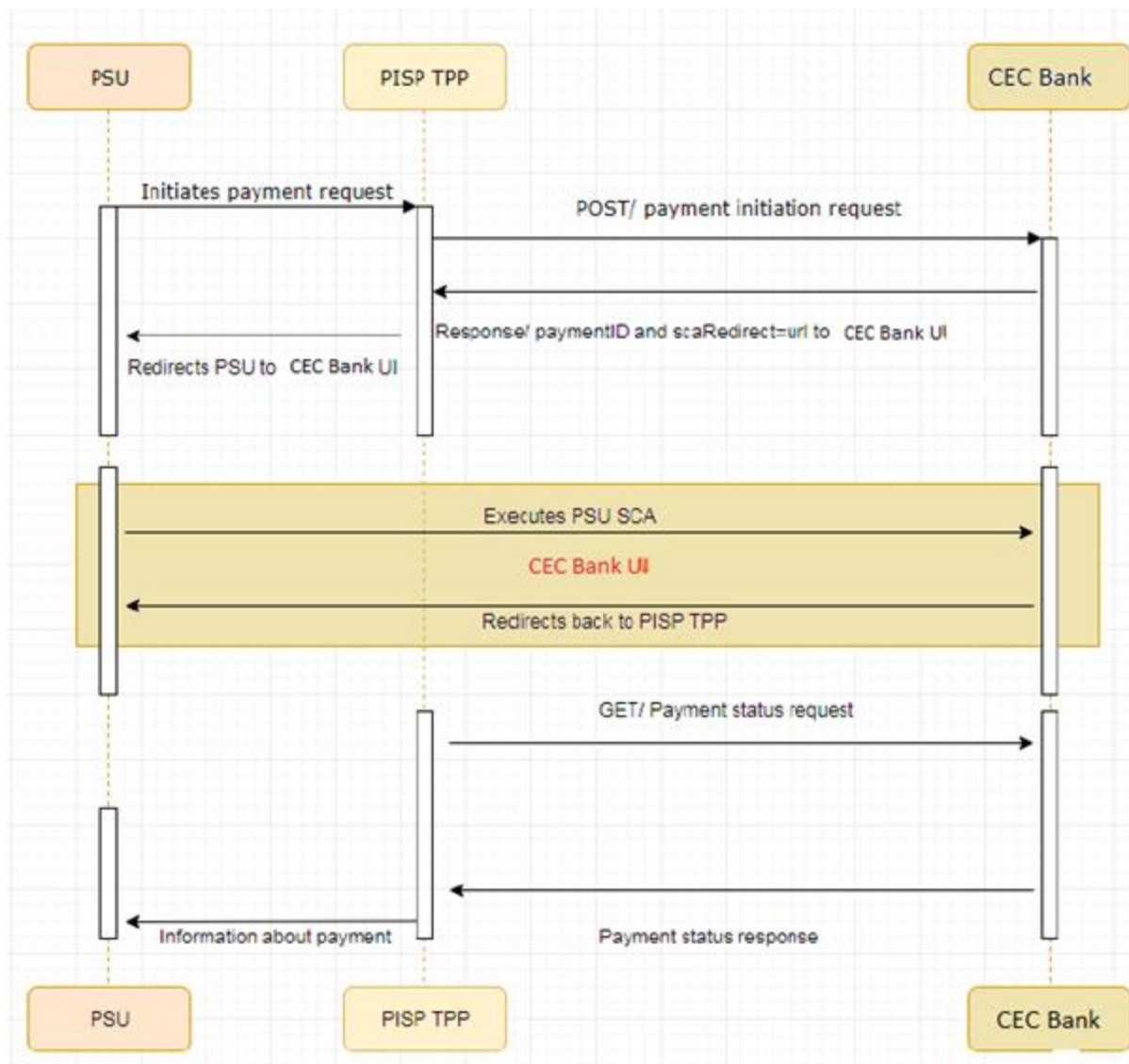
```

2.2 Initiate and authorize payments using PSD2 API-PaymentsC service operations

The PSD2 API-PaymentsC service can be used by TPP with PISP role (EIDAS certificate that includes the PSP_PI role) in order to initiate payments on behalf of a customer.

It is assumed that the data about the debtor account involved in the transaction has been obtained previously, usually by using the GET /v1/accounts operation in the PSD2 API-AccountsC service.

The payment initiation/processing flow diagram is shown in the figure below:



The following steps are distinguished in the flow:

1. The (PSU) customer requests the payment initiation in the TPP application. The TPP application uses the POST operation `/v1/{payment-service}/{payment-product}` to transmit to the Bank the payment data requested by the customer.
2. The Bank service verifies the correctness of the data and stores the payment data, the response returned to the TPP application specifying the payment identifier and the scaOAuth link (used to determine the URL of the interface exposed by the bank for payment authorization).
3. The TPP application redirects the customer to the authorization interface exposed by the bank.

4. In the CEC Bank Authorization Server interface, the customer is authenticated in the bank's system based on the login credentials in the Internet Banking-CEC Online application.
5. After successful authentication, the customer agrees to process the payment or can reject it.
6. The customer is redirected to the TPP application. If the payment has been authorized by the customer, a code is sent to the redirect address of the TPP application, which can be used by the TPP application to obtain the token to access the payment status/data.
7. The TPP application obtains tokens to access data and payment status. To check the payment status use the GET /v1/{payment-service}/{payment-product}/{paymentId}/status operation.
8. The bank service returns the payment status to the TPP application, which displays it to the customer.

Example of flow:

2.2.1 POST operation

/v1/{payment-service}/{payment-product} can be used with the values *payment-service=payments* and *payment-product=sepa-credit-transfer*, and the payment data is sent in JSON format. The parameters TPP-Redirect-URI, X-IBM-Client-Id, X-Request-ID and TPP-Signature-Certificate in the request header are mandatory.

The debtorAccount parameter is optional in the flow. You can choose the option to introduce in the request body or select it on the authorization page if it is not sent in the body.



PSD2 API-PaymentsC

2.0.0

● Overview

- POST /v1/{payment-ser...
- GET /v1/{payment-service}/...
- DELETE /v1/{payment-servi...
- GET /v1/d/{payment-servic...
- DELETE /v1/d/{payment-se...
- GET /v1/{payment-service}/...
- GET /v1/d/{payment-servi...
- POST /v1/{payment-service...
- GET /v1/{payment-service}/...
- GET /v1/{payment-service}/...
- PUT /v1/{payment-service}/...
- POST /v1/{payment-service...
- GET /v1/{payment-service}/...
- GET /v1/{payment-service}/...
- PUT /v1/{payment-service}/...
- POST /v1/funds-confirmati...
- POST /v1/d/funds-confirma...
- Definitions

initiatePayment

DETAILS

TRY IT

POST

Production, <https://api-test.cec.ro/cec/tpp/psd29p/v1/{payment-service}/{payment-Development:product}>

Security

Identification

API Key

Copy client ID

TestAPP - Crede

Only subscribed credentials are selectable.

Parameters

Header

Accept

application/json

Content-Type

application/json

● Definitions



application/json

Content-Type

application/json

X-Request-ID

Generate

c5271314-0f22-5fe4-89fd-9f6c9efb23dd

PSU-IP-Address

Generate

251.172.186.118

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This mi...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

PSU-ID

Generate

Client ID of the PSU in the ASPSP client interface. Might be mand...



Path ^

payment-service*

payments



payment-product*

sepa-credit-tran



Body ^

body

Generate

generated

```
{
  "instructedAmount": {
    "currency": "RON",
    "amount": "3.50"
  }
}
```

RESET

SEND

Response

✓ 201 Created

Body

Headers

```
{
  "transactionStatus": "RCVD",
  "paymentId": "d9a7b868-b3e0-11ef-9a7b-0a0800410000",
  "_links": {
    "scaOAuth": {
      "href": "https://api-test.cec.ro/cec/tpp/oauth/.well-known/oauthauthorization-server"
    },
    "self": {
      "href": "v1/payments/sepa-credit-transfers/d9a7b868-b3e0-11ef-9a7b-0a0800410000"
    },
    "status": {
      "href": "v1/payments/sepa-credit-transfers/d9a7b868-b3e0-11ef-9a7b-0a0800410000/status"
    }
  }
}
```

The response indicates the status "RCVD" (transaction registered for authorization), the payment reference-paymentId, and the scaOAuth link to be used to determine the payment authorization endpoint, based on the Oauth 2.0 authorization server metadata.

2.2.2 The TPP application must redirect the PSU to the bank login page

using the authorization endpoint provided by the scaOAuth link.



```
{
  "issuer": "https://api-test.cec.ro",
  "authorization_endpoint": "https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/authorize",
  "token_endpoint": "https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/token",
  "response_types_supported": [
    "code",
    "none"
  ],
  "response_modes_supported": [
    "fragment",
    "form_post"
  ],
  "grant_types_supported": [
    "authorization_code"
  ],
  "id_token_signing_alg_values_supported": [
    "RS256"
  ],
  "id_token_encryption_alg_values_supported": [
    []
  ],
  "id_token_encryption_enc_values_supported": [
    []
  ],
  "poc": "https://api-test.cec.ro/cec/",
  "name": "OauthCEC",
  "registration_endpoint": "https://apiportal-test.cec.ro/cec/tpp/"
}
```

The example of the URL formation for Redirect can be found in the "Location" attribute of the response header of the POST operation /v1/{payment-service}/{payment-product}:

Location :

<https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/authorize?>

response_type=code

&scope=PIS:5d935a04-7756-11eb-a2f7-0a06001a0000

&redirect_uri=https://example.com/redirect

&client_id=286xxxxxxxxxxxxxxxxxxfe8a3

&code_challenge=<code challenge value>

2.2.3 The TPP application redirects the customer to the URL formed in point 2.2.2

which opens the customer login screen in the bank system. The customer is authenticated by the bank based on the user defined in the Internet Banking-CECOnline application and the access code generated by the physical device he/she owns or by the eToken application installed on the mobile phone

api-test.cec.ro/cec/tpp/oauthcec/oauth2/authorize?response_type=code&scope=PIS:5d935a04-7756-11eb-a2f7-0a06001a0000&redirect_uri=https://ex

CEC Bank

CEC Bank - Authorization Platform

User CECOnline
testcecap

Cod acces

Log in

Introduceti Numele dvs de utilizator din aplicatia CECOnline si Codul de acces generat de dispozitivul de

2.2.4 If successfully logged in, the customer can view the initiated payment data from the TPP application.

If the IBAN has been transmitted in the <debtorAccount> parameter it is displayed in the <Iban Account Payer> area.(see figure below)



CEC Bank

**Confirmare Plata din cont deschis la CEC Bank
initiată din aplicația "AppTestTpp1"**

Pentru a confirma plata cu datele de mai jos acționați "Confirm plata!".

Identificator plata	5d935a04-7756-11eb-a2f7-0a06001a0000
Suma de plata	100.00 RON
Iban Cont platitor	RO42CECEQW9898bON9993644
Denumire Beneficiar	Creditor Name
Iban Cont beneficiar	RO51RZBR7531710377217771433
Detalii plata	test description

Refuz plata! **Confirm plata!**

Puteti autoriza plata numai daca sunteti utilizator al serviciului Internet Banking-CECOnline.

If the <debtorAccount> parameter has not been filled in when initiating the payment, in the authorization window you can select from a list. The list is filled in with all the IBANs from which the payment can be made (see figure below)

CEC Bank

Confirmare Plata din cont deschis la CEC Bank
initiată din aplicația "Sandbox dev"

Pentru a confirma plata cu datele de mai jos acționați "Confirm plata!".

Identificator plata	431fb6b2-e059-11ee-9b61-0a1200740000
Suma de plata	3.50 RON
Iban Cont platitor	RO42CECEQW9898bON9993644
Denumire Beneficiar	RO42CECEQW9898bON9993644 RO49CECER79292bON8883343
Iban Cont beneficiar	DE02100100109307118603
Detalii plata	Ref Number Merchant

Refuz plata!

Confirm plata!

Puteti autoriza plata numai daca sunteti utilizator al serviciului Internet Banking-CECOnline.

You can authorize the payment by pressing the "Confirm payment!" button or cancel it by pressing the "Reject payment!" button

api-test.cec.ro says

Plata a fost trimisa catre procesare. Pentru mai multe informatii, verificati statusul platii.

OK

Confirmare Plata din cont deschis la CEC Bank
initiată din aplicația "AppTestTpp1"

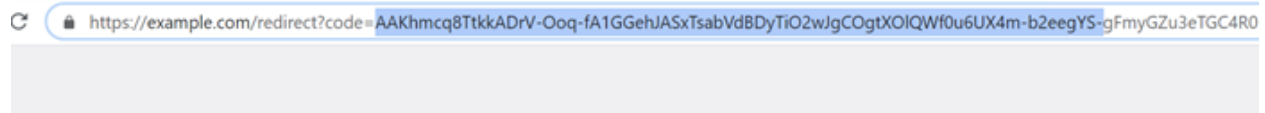
Pentru a confirma plata cu datele de mai jos acționați "Confirm plata!".

Identificator plata	5d935a04-7756-11eb-a2f7-0a06001a0000
Suma de plata	100.00 RON
Iban Cont platitor	RO42CECEQW9898bON9993644
Denumire Beneficiar	Creditor Name
Iban Cont beneficiar	RO51RZBR7531710377217771433
Detalii plata	test description

Refuz plata!

Confirm plata!

2.2.5 If the payment has been authorized by the customer, a code is sent to the redirect address of the TPP application, which can be used by the TPP application to obtain the token to access the payment status.



2.2.6 The TPP application will provide the above access code in exchange for a token to access the payment data (Bearer Token)

by calling the POST method `/oauth2/token` from the URL indicated in the metadata "token_endpoint", with the purpose "PIS:paymentId" (same purpose contained in the access code provided in the previous step)

The request body must be sent in *x-www-form-urlencoded* format and must contain the parameters below (the values shown correspond to the in/out parameters of the authorization operation presented in the previous step):

- grant_type=authorization_code
- client_id=28xxxxxxxxxxfe8a3
- code= AAKhmcq8TtkkADrV-Ooq-fA1GGehJASxTsabVdBDyTiO2wJgCOgtXOIQWf0u6UX4m-b2eegYS-gFmyGZu3eTGC4R0QGZls6fLVI2XrKjsgJlqwE9gd6cF5YxsxgLqWyCoU8sg_MhhedaRianFHhD6JPF
- redirect_uri= <https://example.com/redirect>
- scope= PIS:5d935a04-7756-11eb-a2f7-0a06001a0000
- code_verifier=ginfeftejiscu

200 OK

```
{ "token_type": "bearer",
```



```

"access_token":
"AAIkMjg2Yjk1NmYtZDQxMS00OTM4LTljMGUtYmIODE1YWZlOGZckcP8z5nf6w2H6
KjCNXCtdGSONM3-
wBIrrZcI6kSrgUMFmHWWeJL7jJ6V6TyF4IPrDJ5j7bYt_LBiRxZW_YkFIoN2qn1mHlqEjFFb
Epy2PxFpCn5RgruyLBdc8r5-0vnAB5q2ThhRWQzw8LJXIKw",

"expires_in": 3595,

"consented_on": 1614253740,

scope": "PIS:5d935a04-7756-11eb-a2f7-0a06001a0000",

"refresh_token": "AAJE8ZA8occf5oiHN6ImOYwIwLaTkXDVv9jO1xyNDe3JMRVXMOd8e-
Q0Wfu0Q5TOLvDa5OW_4UqiMrhLSDa79xiwI-
7kplVeT_a57AclheDIftRzR1eDWJYoHan34yutNrXMU7KY3XMe_rItRIQXSEzbs98FiMc_jU5
AlklZPXGQw",

"refresh_token_expires_in": 2682000

}

```

In the response of the POST /oauth2/token operation, the access token value needed to call the other operations in the PSD2 API API-PaymentsC API is found. It also specifies the refresh_token value, with a validity period corresponding to the validity of the consent (max. 90 days), which can be used to generate a new access_token without PSU involvement.

To get a new valid access_token in exchange for the refresh_token, use the same POST /oauth2/token operation, but with grant_type=refresh_token and the required client_id and refresh_token parameters, as in the example below.


```


curl --request POST \
--url https://api-test.cec.ro/cec/tpp/oauthcec/oauth2/token \
--header 'accept: application/json' \
--header 'content-type: application/x-www-form-urlencoded' \
--header 'x-ibm-client-id: 28xxxxxxxxxxfe8a3\
--data 'grant_type=refresh_token&client_id=28xxxxxxxxxxfe8a3&refresh_token
AAJE8ZA8occf5oiHN6ImOYwIwLaTkXDVv9jO1xyNDe3JMRVXMOd8e-
Q0Wfu0Q5TOLvDa5OW_4UqiMrhLSDa79xiwI-
7kplVeT_a57AclheDIftRzR1eDWJYoHan34yutNrXMU7KY3XMe_rItRIQXSEzbs98FiMc_jU5
AlklZPXGQw

```

2.2.7 Payment status

You get it with GET `/v1/{payment-service}/{payment-product}/{paymentId}/status` . In the Authorization parameter of the header, enter the value `access_token` obtained in step 7.



PRODUCTS v HOW IT WORKS CONTACT US EN 

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-PaymentsC - API Developer Portal

PSD2 API-PaymentsC


2.0.0


- Overview
- POST /v1/{payment-service}...
- GET /v1/{payment-service}/...
- DELETE /v1/{payment-servi}...
- GET /v1/d/{payment-service}...
- DELETE /v1/d/{payment-se}...
- GET /v1/{payment-servi}...
- GET /v1/d/{payment-service}...
- POST /v1/{payment-service}...
- GET /v1/{payment-service}/...
- GET /v1/{payment-service}/...
- PUT /v1/{payment-service}/...
- POST /v1/{payment-service}...
- GET /v1/{payment-service}/...
- GET /v1/{payment-service}/...
- PUT /v1/{payment-service}/...
- POST /v1/funds-confirmati...
- POST /v1/d/funds-confirma...
- Definitions


getPaymentInitiationStatus

DETAILS TRY IT

GET Production, `https://api-test.cec.ro/cec/tpp/psd29p/v1/{payment-service}/{payment-Developmentproduct}/{paymentId}/status`

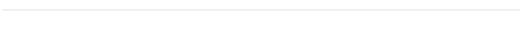

Security `oauth` 


Identification 

API Key `TestAPP - Crede`  [Copy client ID](#)

Only subscribed credentials are selectable.

API Secret



Authorization 



Parameters

Header ^

Accept

application/json

Content-Type

application/json

X-Request-ID

Generate

82ccc8d7-92f1-58dd-9e6e-57d22f332a02

Digest

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

A signature of the request by the TPP on application level. This mi...

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

PSU-IP-Address

Generate

38.122.104.38

PSU-IP-Port

Generate



PSU-Geo-Location

Generate

The forwarded Geo Location of the corresponding http request b...

Path ^

payment-service*

payments ▾

payment-product*

sepa-credit-tran ▾

paymentId*

Generate

4895873435697152

RESET

SEND

Response

✓ 200 OK

Body

Headers

```
{  
  "transactionStatus": "ACCP"  
}
```



Payment status can be (ISO 20022):

'RCVD': 'Received' - Payment initiation has been received by the receiving agent (status before payment authorization by the customer)

'ACCP': 'AcceptedCustomerProfile' - Preceding check of technical validation was successful. Customer profile check was also successful(payment status after customer authorization).

'ACTC': 'AcceptedTechnicalValidation' - Authentication and syntactical and semantical validation are successful(payment technically verified).

'ACSP': 'AcceptedSettlementInProgress' - All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution (payment in progress)

'ACSC': 'AcceptedSettlementCompleted' - Settlement on the debtor's account has been completed (the amount to pay has been debited from the payer's account)

'ACCC': 'AcceptedSettlementCompleted' - Settlement on the creditor's account has been completed (the amount paid has been registered to the creditor's account)


'RJCT': 'Rejected' - Payment initiation or individual transaction included in the payment initiation has been rejected.(payment not authorized by the customer)

'CANC': 'Cancelled'-Payment initiation has been cancelled before execution(payment refused by the bank for various reasons: lack of availability, AML/Embargo restrictions, etc.).


2.3 Funds confirmation service - PIIS functionality

The operation that can be used to confirm the existence of an amount in the account is POST /v1/funds-confirmations and is developed in the PSD2 API-PaymentsC API.

When calling this operation, a valid bearer access token is requested for the purpose "AIS:consented", where the consent identifier (consented) identifies a valid consent given by the customer to consult the balance of the respective account.



PRODUCTS ▾HOW IT WORKSCONTACT US

EN ▾

Back | API Products | PSD2-CEC-APIC - API Developer Portal | PSD2 API-PaymentsC - API Developer Portal

PSD2 API-PaymentsC

2.0.0

- Overview
- POST /v1/payment-service...
- GET /v1/payment-service/...
- DELETE /v1/payment-servi...
- GET /v1/d/payment-servi...
- DELETE /v1/d/payment-se...
- GET /v1/payment-service/...
- GET /v1/d/payment-servi...
- POST /v1/payment-service...
- GET /v1/payment-service/...
- GET /v1/payment-service/...
- PUT /v1/payment-service/...
- POST /v1/payment-service...
- GET /v1/payment-service/...
- GET /v1/payment-service/...
- PUT /v1/payment-service/...
- POST /v1/funds-confir...**
- POST /v1/d/funds-confirma...
- Definitions

checkAvailabilityOfFunds

DETAILSDetailsTRY IT

POST

Production, <https://api-test.cec.ro/cec/tpp/psd29p/v1/funds-confirmations>
Development:

Parameters

Header

Accept

application/json

Content-Type

application/json

X-Request-ID*

Generate

a8e50151-ff46-5914-b0b9-9f983c8596c1

Consent-ID

Generate

This data element may be contained, if the payment initiation tra...

<PUBLIC>



Signature

Generate

Is contained if and only if the "Signature" element is contained in ...

Signature

Generate

TPP-Signature-Certificate

Generate

-----BEGIN CERTIFICATE-----xxxEXAMPLExxxxxxxxxxEXAMPLExxxxx...

Body

body*

Generate

generated

```
{
  "cardNumber": "6976843640995840",
  "account": {
    "iban": "FR7612345987650123456789014",
    "bban": "BARC12345612345678"
  }
}
```

RESET

SEND

If the access token is not fit for purpose or has expired or the consent resource does not include the account for which confirmation is requested, error codes 401 (UNAUTHORIZED) and 403 (FORBIDDEN) are returned.

2.4 Initiate and authorize bulk payments using PSD2 API-PaymentsC service operations

The following steps are distinguished in the flow:

2.4.1 The (PSU) customer requests the payment initiation in the TPP application.

The TPP application uses the POST operation **<https://api-test.cec.ro/tpp/psd29p/v1/bulk-payments/sepa-credit-transfer>** to transmit to the Bank the payment data requested by the customer.

The image below shows an example of a call using POSTMAN and the body used:

POST

https://api-test.cec.ro/cec/tpp/psd29p/v1/bulk-payments/sepa-credit-transfers

Params

Authorization

Headers (14)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

```

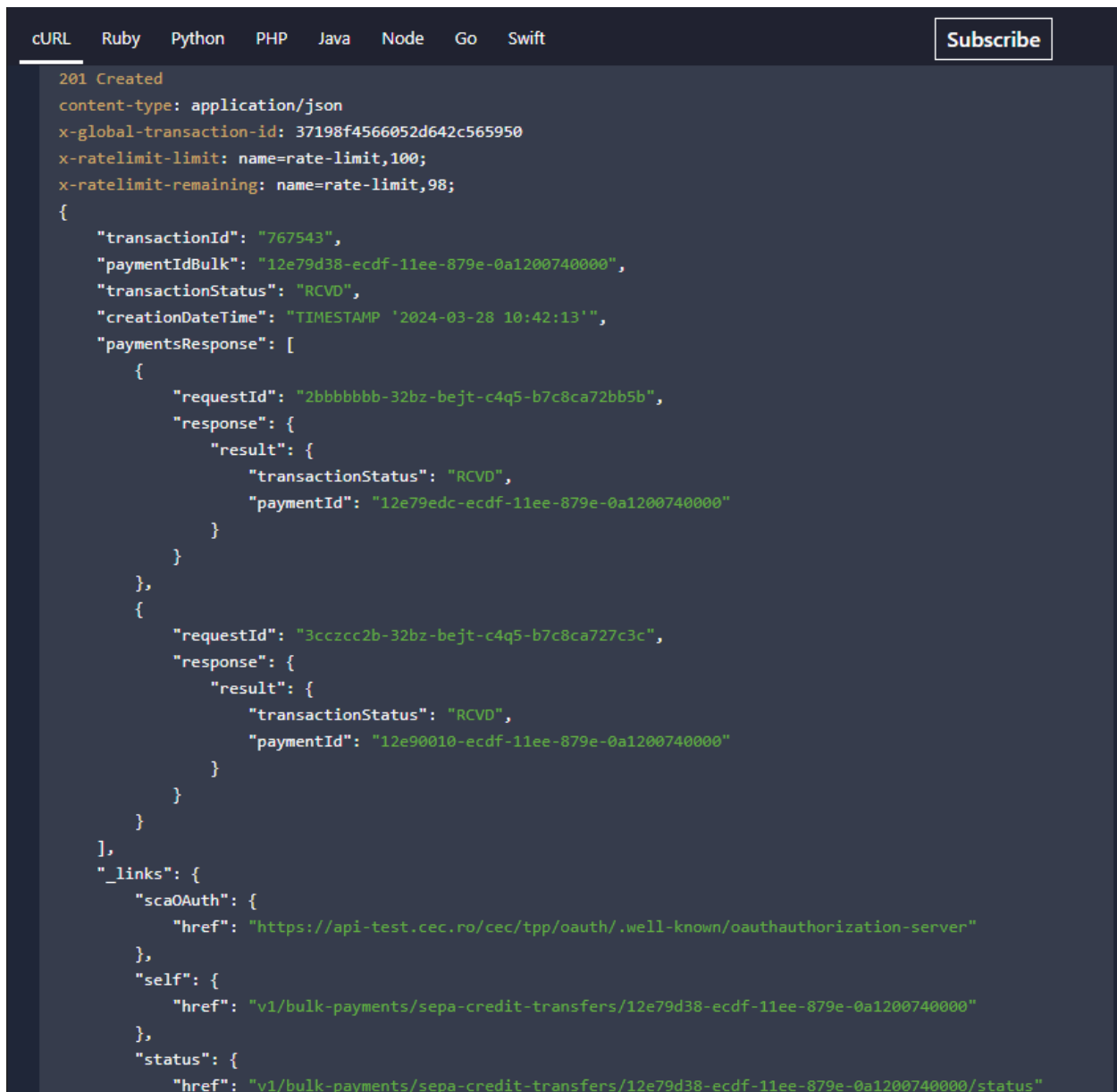
1  {
2    "debtorAccount": {
3      "iban": "RO30CECEB002003765290100",
4      "debtorName": "Ion Popescu",
5      "endToEndId": "767543"
6    },
7    "paymentExecutionType": "instant",
8    "requestedExecutionDate": "tbd",
9    "payments": [
10     {
11       "requestId": "2bbbbbbb-32bz-bejt-c4q5-b7c8ca72bb5b",
12       "instructedAmount": {
13         "currency": "RON",
14         "amount": "1"
15       },
16       "creditorName": "Ion Popescu 10",
17       "creditorAccount": {
18         "iban": "RO59RNCB0857169988050001"
19       },
20       "remittanceInformationUnstructured": "test plata bulk cont BCR"
21     },
22     {
23       "requestId": "3cczcc2b-32bz-bejt-c4q5-b7c8ca727c3c",
24       "instructedAmount": {
25         "currency": "RON",
26         "amount": "1"
27       },
28       "creditorName": "Ion Popescu 11",
29       "creditorAccount": {
30         "iban": "RO82RZBR0000060023923579"
31       },
32       "remittanceInformationUnstructured": "test plata bulk cont Raiffeisen"
33     }
34   ]
35 }

```

2.4.2 The Bank Service checks the correctness of the data and stores the payment data

In the response returned to the TPP application specifying the payment identifier and the scaOAuth link (used to determine the URL of the interface exposed by the bank for payment authorization).

In the figure below there is an example of the response to initiate the payment if using the developer portal <https://apiportal-test.cec.ro/cec/tpp>



```
201 Created
content-type: application/json
x-global-transaction-id: 37198f4566052d642c565950
x-ratelimit-limit: name=rate-limit,100;
x-ratelimit-remaining: name=rate-limit,98;
{
  "transactionId": "767543",
  "paymentIdBulk": "12e79d38-ecdf-11ee-879e-0a1200740000",
  "transactionStatus": "RCVD",
  "creationDateTime": "TIMESTAMP '2024-03-28 10:42:13'",
  "paymentsResponse": [
    {
      "requestId": "2bbbbbbb-32bz-bejt-c4q5-b7c8ca72bb5b",
      "response": {
        "result": {
          "transactionStatus": "RCVD",
          "paymentId": "12e79edc-ecdf-11ee-879e-0a1200740000"
        }
      }
    },
    {
      "requestId": "3cczcc2b-32bz-bejt-c4q5-b7c8ca727c3c",
      "response": {
        "result": {
          "transactionStatus": "RCVD",
          "paymentId": "12e90010-ecdf-11ee-879e-0a1200740000"
        }
      }
    }
  ],
  "_links": {
    "scaOAuth": {
      "href": "https://api-test.cec.ro/cec/tpp/oauth/.well-known/oauthauthorization-server"
    },
    "self": {
      "href": "v1/bulk-payments/sepa-credit-transfers/12e79d38-ecdf-11ee-879e-0a1200740000"
    },
    "status": {
      "href": "v1/bulk-payments/sepa-credit-transfers/12e79d38-ecdf-11ee-879e-0a1200740000/status"
    }
  }
}
```

2.4.3 The TPP application redirects the customer to the authorization interface exposed by the bank

which is the same as a single payment. In the CEC Bank Authorization Server interface, the customer is authenticated in the bank's system based on the login credentials in the Internet Banking-CEC Online application.

- 2.4.4 After successful authentication, the customer agrees to process the bulk payment or can reject it.

CEC Bank

Confirmare Plata din cont deschis la CEC Bank
initiată din aplicația

Iban Platitor

Pentru a confirma plata cu datele de mai jos acționați "Confirm plata!"

Suma de plata	Denumire Beneficiar	Iban Beneficiar	Detalii Plata
1.00 RON	Ion Popescu 10	RO59RNCB0857169988050001	test plata bulk cont BCR
1.00 RON	Ion Popescu 11	RO82RZBR0000060023923579	test plata bulk cont Raiffeisen

Refuz plata!

Confirm plata!

Puteti autoriza plata numai daca sunteti utilizator al serviciului Internet Banking-CECOnline.

- 2.4.5 The customer is redirected to the TPP application.

If the bulk payment has been authorized by the customer, a code is sent to the redirect address of the TPP application, which can be used by the TPP application to obtain the token to access the payment status/data.

2.4.6 The TPP application obtains tokens to access data and bulk payment status.

To check bulk payment status use the GET operation <https://api-test.cec.ro/cec/tpp/psd29p/v1/d/{payment-service}/{payment-product}/{paymentId}>

The bank service returns the status of the bulk payment to the TPP application, which displays it to the customer (see figures below)

The figure below shows that the bulk payment has the status "ACCP".

```
200 OK
content-type: application/json
x-global-transaction-id: 37198f4566052e981e69bf91
x-ratelimit-limit: name=rate-limit,100;
x-ratelimit-remaining: name=rate-limit,93;
{
  "transactionId": "767543",
  "paymentIdBulk": "12e79d38-ecdf-11ee-879e-0a1200740000",
  "transactionStatus": "ACCP",
  "creationDateTime": "TIMESTAMP '2024-03-28 10:42:13'",
  "debtorAccount": {
    "iban": "RO30CECEB002003765290100"
  },
  "paymentsResponse": [
    {
      "requestId": "2bbbbbbb-32bz-bejt-c4q5-b7c8ca72bb5b",
      "response": {
        "result": {
          "transactionStatus": "ACCP",
          "paymentId": "12e79edc-ecdf-11ee-879e-0a1200740000",
          "creditorAccount": {
            "iban": "RO59RNCB0857169988050001"
          },
          "creditorName": "Ion Popescu 10",
          "instructedAmount": {
            "currency": "RON",
            "amount": "1.00"
          },
          "remittanceInformationUnstructured": "test plata bulk cont BCR"
        }
      }
    },
    {
      "requestId": "3cczcc2b-32bz-bejt-c4q5-b7c8ca727c3c",
      "response": {
        "result": {
          "transactionStatus": "ACCP",
          "paymentId": "12e90010-ecdf-11ee-879e-0a1200740000"
```

Following the execution of the bulk payments the status can be changed to "PART" following the failed execution of a single bulk payment as in the figure below:

```
Response

200 OK
content-type: application/json
x-global-transaction-id: 37198f4566052eff2d9c77b2
x-ratelimit-limit: name=rate-limit,100;
x-ratelimit-remaining: name=rate-limit,92;
{
  "transactionId": "767543",
  "paymentIdBulk": "12e79d38-ecdf-11ee-879e-0a1200740000",
  "transactionStatus": "PART",
  "creationDateTime": "TIMESTAMP '2024-03-28 10:42:13'",
  "debtorAccount": {
    "iban": "RO30CECEB002003765290100"
  },
  "paymentsResponse": [
    {
      "requestId": "2bbbbbbb-32bz-bejt-c4q5-b7c8ca72bb5b",
      "response": {
        "result": {
          "transactionStatus": "ACCP",
          "paymentId": "12e79edc-ecdf-11ee-879e-0a1200740000",
          "creditorAccount": {
            "iban": "RO59RNCB0857169988050001"
          },
          "creditorName": "Ion Popescu 10",
          "instructedAmount": {
            "currency": "RON",
            "amount": "1.00"
          },
          "remittanceInformationUnstructured": "test plata bulk cont BCR"
        }
      }
    },
    {
      "requestId": "3cczcc2b-32bz-bejt-c4q5-b7c8ca727c3c",
      "response": {
        "result": {
```

```

{
  "requestId": "3cczcc2b-32bz-bejt-c4q5-b7c8ca727c3c",
  "response": {
    "result": {
      "transactionStatus": "CANC",
      "paymentId": "12e90010-ecdf-11ee-879e-0a1200740000",
      "creditorAccount": {
        "iban": "RO82RZBR0000060023923579"
      },
      "creditorName": "Ion Popescu 11",
      "instructedAmount": {
        "currency": "RON",
        "amount": "1.00"
      },
      "remittanceInformationUnstructured": "test plata bulk cont Raiffeisen"
    }
  }
}
1
}

```

NOTE:

PART, ACSC - final statuses for a bulk payment initiation transaction // PART status is provided in Berlin Group

Any other status up to this point is an intermediate status.

PART - at least one bulk payment was not successfully completed.

ACSC- all bulk payments have been successfully completed.

PATC - payment requires additional signatures