

Manual

TITLE	Documentation for TPP access under PSD2 RTS art 33 §5
VERSION	1
CREATION	2019-12-11
AUTHOR	IT departement
DEPARTMENT	IT

DOCUMENT CONTROL INFORMATION

DIFFUSION	Document public
REVIEW CYCLE	None

REVIEW HISTORY	BY
2019-12-11	IT departement

APPROBATIONS	BY
2019-12-12	Authorised Management

SUMMARY

1. PURPOSE OF THIS DOCUMENT	3
2. GENERAL INTERFACE AND TESTING FACILITY.....	3
3. AUTHENTICATION OF THE TPP.....	3
4. RETRIEVAL OF INFORMATION	3
4.1 DATA LOCATION ON A PAGE	3
4.2 ACCOUNTS LIST	4
4.3 MOVEMENTS LIST	6
4.4 BENEFICIARIES LIST.....	7
5. INITIATION OF TRANSACTIONS.....	9
5.1 INTERN TRANSFER	9
5.2 SEPA TRANSFER.....	11
5.3 SWIFT TRANSFER.....	12

1. PURPOSE OF THIS DOCUMENT

This document aims to clarify how Third Party Provider, in the sense of PSD2, can connect to the general interface made available to the payment service user.

TPP can connect to the general interface in the absence of a dedicated interface, or to use the contingency mechanism as described in article 33 §4 of the COMMISSION DELEGATED REGULATION (EU) 2018/38¹.

2. GENERAL INTERFACE AND TESTING FACILITY

The general interface is available at <https://ebanking.fortuna.bank/fortuna/>

A testing facility is made available at <https://ebanking.fortuna.bank/fortuna/demo/>

3. AUTHENTICATION OF THE TPP

To allow the proper identification of the TPP², they should add an HTTP header “Signature” in accordance with the draft RFC “Signing HTTP Messages”³.

4. RETRIEVAL OF INFORMATION

In order to ease information retrieval for the TPP, the relevant data are placed in each page in a machine-readable format.

For each case a “key” is required, it has to be noted that the encoded value is unique to the session of the user, and will be different the next time the user connects to the service.

4.1 DATA LOCATION ON A PAGE

In each section of this document, the data to be retrieved by a TPP is located in a hidden div with the

```
Id "tpp": <div id="tpp" style="display: none"> {JSON string} </div>
```

Then, the data itself is represented as a JSON string.

¹ <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32018R0389>

² Article 33 §5: “the payment service providers referred to in Article 30(1) can be identified”

³ <https://tools.ietf.org/html/draft-cavage-http-signatures-12>

4.2 ACCOUNTS LIST

URL: /fortuna/banking

JSON format:

"accountKey":

"stringValue": Encoded string representing the account (this string is unique in the session)

i.e:

"ea55b45a703a15b291a3eb7de659e97a4fa734c961dde291bd57957e59376708"

"databaseValue": Uncoded string referring to the account number

i.e: "1244699781001001"

"accountType":

"stringValue": internal denomination of the account

i.e: "FBSC.ACCOUNTS"

"cash": boolean telling if it is a cash account

i.e: "true"

"saving": boolean telling if it is a saving account

i.e: "false"

"iban":

"stringValue": string representin the IBAN of the account

i.e: "LU123412341324123412"

"fortuna": boolean telling if the IBAN is a Fortuna IBAN (internal account)

i.e: "true"

"valid": boolean telling if the IBAN is valid

i.e: "true"

"userLabel": string of the internal name type of the account

i.e: "Compte courant vert"

"accountCurrency":

"currencyCode": string of the currency

i.e: "EUR"

"stringValue": string of the currency

i.e: "EUR"

"accountBalance": decimal representing the account balance

i.e: -150.52

i.e: 10000.00

"creditRate": decimal representing the credit rate of the account

i.e: 0.01

i.e: null

"debitRate": decimal representing the debit rate of the account

i.e: 0.01

i.e: null

"standingOrderCapable": boolean telling if the account can make standing order

i.e: "true"

"directDebitCapable": boolean telling if the account can have direct debit

i.e: "true"

"transferCapable": boolean telling if the account can make transfer

i.e: "true"

"externalTransferCapable": boolean telling if the account can make external transfer

i.e: "true"

"ownTransferCapable": boolean telling if the account can make transfer to an other owner's account

i.e: "true"

"transferTargetCapable": boolean telling if this account may be the target of transfers

i.e: "true"

"movementCapable": boolean telling if this account may have movements

i.e: "true"

"owners": list of string containing the name of the owner(s)

i.e: ["John DOE"]

i.e: ["John DOE", "Jane DOE"]

"productGroup": string representing the internal denomination of the account

i.e: "FBSC.ACCOUNTS"

"labelCode": string representing the internal denomination of the account

i.e: "FBSC.ACCOUNTS"

"virtualBalance": decimal representing the virtual balance of the account

i.e: 98.05

"creditLimit": decimal representing the credit limit of the account

i.e: 0.00

i.e: 100000.00

4.3 MOVEMENTS LIST

URL: /fortuna/banking/account/movements.progressivedisplay?t:ac={stringValue of the accountKey}/{true for descending date order, false for ascending date order}/{from date}/{to date}

i.e:/fortuna/banking/account/movements.progressivedisplay?t:ac=ea55b45a703a15b291a3eb7de659e97a4fa734c961dde291bd57957e59376708/true/01\$002f01\$002f2019\$002000:00:00/31\$002f12\$002f2019\$002000:00:00

(movements from 01/01/2019 to 31/12/2019)

HTML Div:

```
<div id="tpp" style="display: none">
```

```
    pastMovements: [ {first movement}, {second movement}, ... ]
```

```
    futureMovements : [ {first movement}, {second movement}, ... ]
```

```
</div>
```

JSON Format:

"accountKey":

"stringValue": Encoded string representing the account (this string is unique in the session)

i.e:

"ea55b45a703a15b291a3eb7de659e97a4fa734c961dde291bd57957e59376708"

"databaseValue": Uncoded string referring to the account number

i.e: "1244699781001001"

"transferKey":

"stringValue": Encoded string representing the movement (this string is unique in the session)

i.e:

"8b36d0d1c6343e67cce68ba431a9f75cd2e10d896a10bcb3051647295dff8ec2"

"databaseValue": Uncoded string referring to the movement

i.e: "FT1928058JTZ"

"label": list of string containing information about the movement

i.e: ["IncomingTransfer", "Joe DOE", "LU1234", "Birthday"]

"valueDate": string representing the value date of the movement

i.e: "2019-01-31"

"amount": decimal representing the amount of the mouvement (amount currency is the currency of the account). A positive amount is a credit and a negative amount is a debit.

i.e: -10.00

i.e: 10.00

"accountDate": string representing the date when the movement appeared in the account

i.e: "2019-01-31"

"balance": decimal representing the balance of the account with the movement took into account

i.e: -102.52

i.e: 10000.00

4.4 BENEFICIARIES LIST

URL : /fortuna/banking/beneficiary

HTML Div:

```
<div id="tpp" style="display: none">  
    [ {first beneficiary}, {second beneficiary}, ... ]  
</div>
```

JSON Format:

"beneficiaryKey":

"databaseValue": Unencoded string representing the ID of the beneficiary

i.e: "BEN9415708397"

"stringValue": Encoded string representing the ID of the beneficiary

i.e: "5cd5242c3a1b6d822f0f0579c7f1b81a"

"beneficiaryOfficialName": string representing the official name of the beneficiary (the one used in the transfer)

i.e: "Mr. John DOE"

"nickName": string representing the nick name of the beneficiary

i.e: "Uncle John",

"transactionCode": string representing the type of transfer the beneficiary is used for.

i.e: "ACSO" (SEPA transfer)

i.e: "AC" (transfer to a fortuna account)

i.e: "OT03" (SWIFT transfer)

"transactionType": string representing the explicit type of transfer the beneficiary is used for.

i.e: "SEPA Credit Transfer Outgoing"

"destBIC": string representing the bic of the beneficiary for a SEPA transfer (ACSO) but name of the bank for a SWIFT transfer (OT03).

i.e: "FOTNLULL"

i.e: ""

"destIBAN":

"stringValue": string representing the IBAN of the beneficiary.

i.e: "LU850019435520057000"

"valid": boolean telling if the iban is valid.

i.e: true

"fortuna": boolean telling if the IBAN is a Fortuna IBAN (internal account)

i.e: "true"

"destAccountNumber": string representing the first line of the SWIFT (could be name of the bank, address of the bank, name of the account, ...).

i.e: "Bank of Beirut"

i.e: "null"

"benAccountNum": string representing the account number of the beneficiary for a SWIFT transfer.

i.e: "1234"

5. INITIATION OF TRANSACTIONS

5.1 INTERN TRANSFER

URL: /fortuna/banking/account/transfer/new/{stringValue of the accountKey}/AC

POST Request: /fortuna/banking/account/transfer/new.inputform

Body :

One future transfer to an existing beneficiary:

beneficiary	{stringValue of the beneficiary} i.e: fd8acc5e250b6f066b2dad7179e26b4
amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup	TRANSFER
date	{date of the transfer execution DD/MM/YYYY} i.e: 31/01/2019
frequency	Monthly

One today transfer to a new beneficiary:

beneficiary	
beneficiaryName	{string for the name of the beneficiary, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: John DOE
iban	{string for the iban} i.e: LU12 0100 3412 3412 341
pseudonyme	{string for the pseudonyme of the beneficiary, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: Uncle John
Amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
Radiogroup	TRANSFER
date	
Frequency	Monthly

One standing order to:

beneficiary	{stringValue of the beneficiary} i.e: fd8acc5e250b6f066b2dad7179e26b4
amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup	STANDING_ORDER
startDate	{date of the standing order start date DD/MM/YYYY} i.e: 31/01/2019
frequency	{string for the frequency of the standing order} i.e: Monthly i.e: Quarterly i.e: Semiannual

	i.e: Annually
untilDateField	{date of the standing order end date DD/MM/YYYY} i.e: 31/12/2019
untilCancelled	{if the standing order period is unlimited, the untilDateField is not applied} value: on

5.2 SEPA TRANSFER

URL: /fortuna/banking/account/transfer/new/{stringValue of the accountKet}/ACSO

POST Request: /fortuna/banking/account/transfer/new.inputform

Body :

One future transfer to an existing beneficiary:

beneficiary	{stringValue of the beneficiary} i.e: fd8acc5e250b6f066b2dad7179e26b4
amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup	TRANSFER
date	{date of the transfer execution DD/MM/YYYY} i.e: 31/01/2019
frequency	Monthly

One today transfer to a new beneficiary:

beneficiary	
beneficiaryName	{string for the name of the beneficiary, 35 alphanum. char. max, special char. allowed: : () ? + , - / . } i.e: John DOE
iban	{string for the iban} i.e: LU12 3412 3412 3412 341
bicCodeACSO	{bic code corresponding to the iban} i.e: FOTNLULL
pseudonyme	{string for the pseudonyme of the beneficiary, 35 alphanum. char. max, special char. allowed:

	: ()?+, -/. } i.e: Uncle John
Amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : ()?+, -/. } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup	TRANSFER
date	
Frequency	Monthly

One standing order:

beneficiary	{stringValue of the beneficiary} i.e: fd8acc5e250b6f066b2dad7179e26b4
amount	{decimal for the amount} i.e: 1000.00
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : ()?+, -/. } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup	STANDING_ORDER
startDate	{date of the standing order start date DD/MM/YYYY} i.e: 31/01/2019
frequency	{string for the frequency of the standing order} i.e: Monthly i.e: Quarterly i.e: Semiannual i.e: Annually
untilDateField	{date of the standing order end date DD/MM/YYYY} i.e: 31/12/2019
untilCancelled	{if the standing order period is unlimited, the untilDateField is not applied} value: on

5.3 SWIFT TRANSFER

URL: /fortuna/banking/

POST Request: /fortuna/banking/account/transfer/new.inputform

Body:

One transfer with an existing beneficiary:

beneficiary	90a9ea416bbe9c4dc13757aa343218d7
radiogroup	WITHOUT_IBAN
radiogroup_0	WITH_BIC
amount	{decimal for the amount} i.e: 1000.00
currency	{string for the currency, ISO 4217} i.e: USD
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup_1	TRANSFER
date	{date of the transfer execution DD/MM/YYYY} i.e: 31/01/2019
frequency	Monthly
fees	{string for the type of fees: shared (SHA), originator (OUR), receiver (BEN)} i.e: SHA i.e: OUR i.e: BEN

Standing order with a new beneficiary

beneficiaryName	{string for the name of the beneficiary, 35 alphanum. char. max, special char. allowed: : () ? + , - / . } i.e: John DOE
radiogroup	WITHOUT_IBAN
accountDest	{string for the account number of the beneficiary, 35 alphanum. char. max, special char. allowed: : () ? + , - / . } i.e: 123456
radiogroup_0	WITHOUT_BIC
bankName	{string for the name of the bank of the beneficiary, 35 alphanum. char. max, special char. allowed: : () ? + , - / . } i.e: Bank X
bankAddress1	{string for the line 1 of the address of the bank, 35 alphanum. char. max, special char. allowed: : () ? + , - / . }

	i.e: 5 rue des fleurs
bankAddress2	{string for the line 2 of the address of the bank, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: 3rd floor
bankAddress3	{string for the line 3 of the address of the bank, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: Luxembourg
pseudonyme	{string for the pseudonyme of the beneficiary, 35 alphanumeric. char. max, special char. allowed: : () ? + , - / . } i.e: John DOE
amount	{decimal for the amount} i.e: 1000.00
currency	{string for the currency, ISO 4217} i.e: USD
destBeneficiaryMessage1	{string for the message1, 35 char. max, special char. allowed: : () ? + , - / . } i.e: bill 12/34 + bil 12/35
destBeneficiaryMessage2	{same as destBeneficiaryMessage1}
destBeneficiaryMessage3	{same as destBeneficiaryMessage1}
destBeneficiaryMessage4	{same as destBeneficiaryMessage1}
radiogroup_1	STANDING_ORDER
startDate	{date of the standing order start date DD/MM/YYYY} i.e: 31/01/2019
frequency	{string for the frequency of the standing order} i.e: Monthly i.e: Quarterly i.e: Semiannual i.e: Annually
untilDateField	{date of the standing order end date DD/MM/YYYY} i.e: 31/12/2019
untilCancelled	{if the standing order period is unlimited, the untilDateField is not applied} value: on
fees	{date of the standing order start date DD/MM/YYYY} i.e: 31/01/2019