| **API - Users** |  |
| --- | --- |
|  |
|  |  |  |  |
| **Version** |  BiB-v2.6  |
| **Reference** | S-money-API-Users |

**Content**

[**1.**](#_heading=h.gjdgxs) **REVISION 2**

[**2.**](#_heading=h.30j0zll) **Resources 3**

[2.1](#_heading=h.1fob9te) User 3

[2.2](#_heading=h.3znysh7) UserProfile 5

[2.3](#_heading=h.2et92p0) SubAccount 5

[2.4](#_heading=h.tyjcwt) BankAccountRef 6

[2.5](#_heading=h.3dy6vkm) CBCardRef 6

[Address 7](#_heading=h.1t3h5sf)

[2.6](#_heading=h.4d34og8) Company 7

[**3.**](#_heading=h.2s8eyo1) **Create a user 7**

[**4.**](#_heading=h.17dp8vu) **Retrieve the list of users 10**

[**5.**](#_heading=h.3rdcrjn) **Retrieve a user 18**

[**6.**](#_heading=h.lnxbz9) **Retrieve a user’S BALANCE 19**

[**7.**](#_heading=h.35nkun2) **Modify a user 20**

[**8.**](#_heading=h.1ksv4uv) **Close the user account 22**

[**9.**](#_heading=h.44sinio) **SEND USER’S ACCEPTED GCU 24**

[**10.**](#_heading=h.2jxsxqh) **Search for a user 25**

[**11.**](#_heading=h.z337ya) **block/UNBLOCK on a user (REVERSIBLE) 27**

[**12.**](#_heading=h.3j2qqm3) **Get a user’s account statement 29**

[**13.**](#_heading=h.1y810tw) **Account statements callback 30**

[**14.**](#_heading=h.4i7ojhp) **Get Account Statements archive 30**

[**15.**](#_heading=h.2xcytpi) **Economic Activity Catalog 31**

# REVISION

| **Version / revision** | **Author** | **Description**  |
| --- | --- | --- |
| BiB-v1.1 |  | 1st version of the doc for BiB |
| BiB-v1.2 | MAROUANIMohamed  | Add “ CountryCode” as mandatory input parameter, in order to identify the Fintech |
| BiB-v1.3 | HAMZA Nabil | Add “Economic activity” as mandatory input parameter, in order to identify the economic activity of the user.Add new errors codes related to KYC specifications. |
| BiB-v1.4 | ROUX Simon | Update §11 Block / Unblock Users §9 (closing account) : Warning has been added |
| BIB-V1.5 | MERAWZA Farouk | Add Delete User Api |
| BIB-V1.6 | MICCOLI Giampiero  | "Create a user" API error table updated (e-mail incorrect, e-mail already in use and city of birth invalid rows) |
| BIB-V1.7 | HAMZA Nabil | Update mandatory fields at user creation |
| BIB-V1.8 | MICCOLI Giampiero  | "Create a user" API:- mandatory fields updated (Civility field)- error table updated (code 5977 added) |
| BIB-V1.9 | Maryem Chebil |  Add Economic Activity Catalog part |
| BIB-V2.0 | Hadrien Lecoq | Characteres authorized for firstname and lastname |
| BIB-V2.1 | MICCOLI Giampiero | Remove “Delete a user” section |
| BIB-V2.2 | TALL Cheikh | Update Format AppUserId X(255) -> X(9)Update Modify a UserUpdate routes for user resource Add KycStatus |
| BiB-V2.3 | Anna GUEYE | Update Search for a User parametersAdd previous / next link to manage pagination |
| BiB-V2.4 | Anna GUEYE | Update Search for a User parameters: add filter hasDebtBalanceAdd “BirthName” as optional input parameter |
| BiB-V2.5 | Anna GUEYE | Add new endpoints for strong user authentication for modify a user and retrive cgu |
| BiB-V2.6 | Anna GUEYE | Update Search for a User parameters: add filter kycStatusAdd new endpoint Get/ Balance to retrieve user’s balance |

# Resources

/api/v[1.x]/users

## User

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Id | Long | N(19) | 114 | S-money user ID |
| AppUserId | String | X(9) | Client-112 | User identifier in the third-party application. 9 characters exactly: [0 - 9 a - z A - Z \_ - .  ! § ( )] |
| CountryCode | String | X(255) | ES | Country Identifiant (**ISO 3166**)ES : EspagneFR : France … |
| Role | int | N(10) | 1 | User’s role1 = Client (Declarative)2 = Extended Client (KYC 2, identity)3 = Extended Client (KYC 3 proof of residency)4 = Extended Client (KYC 4 proof of income) |
| Type | int | N(10) | 1 | User type0 = N/A1 = Individual client2 = Professional client (N/A) |
| Profile | UserProfile |  |  | User’s profile information |
| Amount | Long | N(19) |  | Cumulative Authorization balance of all user sub-accounts (in cents) |
| AccountingBalanceAmout | Long | N(19) |  | Cumulative accounting balance of all user sub-accounts (in cents) |
| SubAccounts | List<Account> |  |  | List of sub-accounts associated with the user |
| BankAccounts | List<BankAccountRef > |  |  | List of the user’s beneficiaries bank accounts |
| CBCards | List<CBCardRef> |  |  | List of user’s bank cards |
| Status | int | N(19) | 1 | User Status1 = OK5 = Closed7 = Blocked |
| Company | Company |  |  | Company name and siret/RNA (Professional account only) |
| KycStatus          | int | N(10) | 0 | Status KYC demand0 = Incomplete1 = Waiting2 = Refused3 = Accepted5: Initialized6 : Complete |
| OnBoardingDate | Datetime | D(YYYY-MM-DDTHH:MM:SS) | 2020-07-29T15:20:31 | User creation date |

## UserProfile

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Civility | Int | N(10) | 0 | Civility 0 = Mr1 = Mrs, Miss |
| FirstName | String | X(140) | Jean | First NameFor card-manufacturing contraints, characters must be : AN or “-“  |
| LastName | String | X(140) | Dupont | Last NameFor card-manufacturing contraints, characters must be : AN or “-“ |
| BirthName | String | X(140) |  |  |
| Birthdate | Datetime | D(YYY-MM-DDTHH:MM:SS) | 1985-09-29T00:00:00 | Birth date |
| Address | Address |  |  | Address of residence |
| Phonenumber | String | X(100) | 0600000001 | Mobile phone number |
| Email | String | X(255) | jeandupont@s-money.fr | E-mail Address |
| Alias | String | X(32) | Jean-dupont | User nickname Maximum number of characters: 32Regex : ^([a-zA-Z])([a-zA-Z0-9\_\-\.\#\s]+)$  |
| Photo | PhotoRef |  |  | Reference to the user photo |
| BirthCity | String | X(255) | Paris | Birth city of the user |
| BirthCountry | String | X(2) | FR | Country of birth of the user (code ISO 3166-1) |
| EconomicActivity | Int |  | 11 | Only specific values of economic activities are allowed. (Code INSEE 2003) |

## SubAccount

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Id | Long | N(19) | 225 | Account identifier |
| AppAccountId | String | X(255) | "123" | Account identifier in the third-party application |
| DisplayName | String | X(140) | Jean Dupont | Display name |
| Amount | Long | N(19) |  | Amount |
| AccountingBalanceAmout | Long | N(19) |  | Authorization balance |
| IsDefault | Bool | B | True | Indicate if it is a main user account |
| CreationDate | DateTime | D(YYY-MM-DDT00:00:00) | 2013-09-29T00:00:00 | Account creation date |
| Iban | String | X(30) | FR1420041010050500013M02606 | Iban of the account |

##  BankAccountRef

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Id | Long | N(19) | 2 | Bank account ID |
| Href | String | X(255) | /api/BankAccounts/2 | Link to bank account details |

## CBCardRef

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Id | Long | N | 12 | Card Id |
| AppCardId | String | X(255) | Card1 | App Card Id |
| Href | String | X(255) | /api/cards/22 | Link to bank card details |

## Address

| **Property** | **Type** | **Format** | **Example** | **Description** |
| --- | --- | --- | --- | --- |
| Street | String | X(255) | 2 rue des châtaigniers | Number and Street |
| ZipCode | String | X(10) | 75001 | Postcode |
| City | String | X(140) | Paris | City |
| Country | String | X(2) | "FR" | Country of residence (code ISO 3166-1) |

## Company

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| Name | String | X(140) | JP-Company | Company or Association name |
| Siret | String | X(255) | 123456789 | Siret or RNA |

# Create a user

To create a user, please provide the following mandatory information:

* AppUserId
* Civility
* CountryCode
* LastName
* BirthName
* FirstName
* BirthDate
* Country (Country of residence)
* City of birth
* Country of birth
* EconomicActivity
* Street
* ZipCode
* City
* PhoneNumber
* Email
* Alias

In order to avoid any confusion, we recommend:

* to use a "companyName-id" type as an AppUserId
* to provide an e-mail Address.

**POST/users**

**Request example:**

POST /api/v[1.x]/users

Authorization: Bearer ACCESS\_TOKEN

**{**

 "AppUserId":"client-112", (mandatory)

 " CountryCode ":"FR", (optional)

 "Profile": (mandatory)

 {

 "Civility":0, (mandatory)

 "FirstName":"Jean", (mandatory)

 "LastName":"Dupont", (mandatory)

 “BirthName”:”” (optional)

 "Birthdate":"1985-09-29T00:00:00", (mandatory)

            "BirthCity": "Paris", (mandatory)
            "BirthCountry": "FR", (mandatory)

 "EconomicActivity" : 11, (mandatory)

 "Address": (mandatory)

 {

 "Street":"2 rue des chataîgners", (mandatory)

 "ZipCode":"75001", (mandatory)

 "City":"Paris", (mandatory)

 "Country":"FR" (mandatory)

 },

 "Phonenumber":"0600000001", (mandatory)

 "Email":"jeandupont@s-money.fr", (mandatory)

"Alias":"jean-dupont" (mandatory)

 }

}

If successful, the above call returns an HTTP code **201**, and returns a JSON structured as follows:

**Response :**

HTTP/1.1 201 CREATED

Content-Type: application/vnd.s-money.v1+json

Location: /api/users/{AppUserId}

{

 "Id":114,

 "AppUserId":"client-112",

 "Role":1,

 "Type":1,

 "Profile":

 {

 "Civility":0,

 "FirstName":"Jean",

 "LastName":"Dupont",

 "Birthdate":"1985-09-29T00:00:00",

 "Address":

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "EconomicActivity" : 11,

 "Phonenumber":"0600000001",

 "Email":"jeandupont@s-money.fr",

"Alias":"jean-dupont"

 },

 "Amount":0,

"AccountingBalanceAmout":0,

 "SubAccounts":

 [{

 "Id":"225",

 "Amount":0,

"AccountingBalanceAmout":0,

 "DisplayName":"Jean Dupont",

 "IsDefault":true,

 "Iban": "FR1420041010050500013M02606"

 }, {…}],

 "BankAccounts":null,

 "CBCards":null,

 "Status":1,

}

**Error cases :**

| **HTTP code**  | **Error code**  | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 400 | 704 | Request parameter not provided |
| 400 | 177 | Invalid value of request parameter |
| 400 | 178 | Invalid request parameter(s) |
| 400 | 567 | Invalid date of birth |
| 400 | 719 | AppUserId or AppAccountId already exists |
| 400 | 377 | The user should be over 18 years old |
| 400 | 312 | e-mail incorrect |
| 400 | 350 | e-mail already in use |
| 400 | 514 | Incorrect or already taken account Alias |
| 400 | 515 | Incorrect or already taken telephone number |
| 400 | 378 | Invalid Country (it needs to be within the EU) |
| 404 | 147 | User cannot be found |
| 400 | 302 | BIC/IBAN not provided, impossible to transfer remaining funds to the user’s account. |
| 400 | 703 | Unitary transaction limit has been reached, impossible to transfer remaining funds to the user’s account |
| 400 | 344 | Annual transaction limit has been reached, impossible to transfer remaining funds to the user’s account |
| 400 | 182 | Unknown Country code |
| 400 | 5977 | User request rejected due for users of American nationality or born in the United States |
| 400 | 6001 | The user should be under 100 years old |
| 400 | 6002 | The user should be over 18 years old in birth country |
| 400 | 6003 | The economic activity must be in Code INSEE 2003 list |
| 400 | 6004 | Unknown Address values (street, zip code, city) |
| 400 | 6005 | City of birth invalid |

# Retrieve the list of users

**Request:**

GET Parameters :

| **Property**  | **Type**  | **Description**  | **Default value**  |
| --- | --- | --- | --- |
| page | Int | Page number (starts with 1) | 1 |
| per\_page | Int | Number of items per page | 50 |

GET /api/v[1.x]/users?page=1&perPage=50

Authorization: Bearer ACCESS\_TOKEN

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "value": [

 {

 "Id": 867,

 "AppUserId": "IF00\_USER05",

 "Type": 1,

 "Role": -1,

 "Profile": {

 "Civility": 0,

 "FirstName": "Jean",

 "LastName": "IF00\_USER05",

 "MaidenName": null,

 "Birthdate": "1979-08-06T00:00:00",

 "Birthcity": "London",

 "BirthCountry": "GB",

 "Address": {

 "Street": "5 rue de la paix",

 "ZipCode": "75017",

 "City": "PAris",

 "Country": "FR"

 },

 "PhoneNumber": "0661618093",

 "Email": "jif00\_user05@gmail.com",

 "Alias": "IF00\_USER05",

 "Picture": {

 "Href": null

 },

 "EconomicActivity": 0,

 "Asset": null,

 "Income": null

 },

 "Credentials": null,

 "Amount": 1000000,

 "AccountingBalanceAmount": 1000000,

 "SubAccounts": [

 {

 "Id": 867,

 "AppAccountId": "IF00\_USER05",

 "Amount": 1000000,

 "AccountingBalanceAmount": 1000000,

 "DisplayName": "Jean IF00\_USER05",

 "CreationDate": "2019-03-08T15:29:05",

 "IsDefault": true,

 "Iban": "FR6716528001020AV000000O381",

 "IbanHint": "FR67XXXXXXXXXXXXXXXXO381"

 }

 ],

 "BankAccounts": [

 {

 "Id": 293,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/293"

 },

 {

 "Id": 1671,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1671"

 },

 {

 "Id": 1685,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1685"

 },

 {

 "Id": 1686,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1686"

 },

 {

 "Id": 1687,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1687"

 },

 {

 "Id": 1711,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1711"

 },

 {

 "Id": 1724,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/bankaccounts/1724"

 }

 ],

 "CBCards": [

 {

 "Id": 1043,

 "AppCardId": "IF00\_USER05\_CP01",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP01"

 },

 {

 "Id": 1044,

 "AppCardId": "IF00\_USER05\_CP02",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP02"

 },

 {

 "Id": 1045,

 "AppCardId": "IF00\_USER05\_CP03",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP03"

 },

 {

 "Id": 1046,

 "AppCardId": "IF00\_USER05\_CP04",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP04"

 },

 {

 "Id": 1047,

 "AppCardId": "IF00\_USER05\_CP06",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP06"

 },

 {

 "Id": 5049,

 "AppCardId": "90d3fe6a7f774a149dd96b",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/90d3fe6a7f774a149dd96b"

 },

 {

 "Id": 5050,

 "AppCardId": "59b9379adb3044af978b69",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/59b9379adb3044af978b69"

 },

 {

 "Id": 5070,

 "AppCardId": "696e7bc8f6654109b22d83",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/696e7bc8f6654109b22d83"

 },

 {

 "Id": 5128,

 "AppCardId": "4b5c3901d1a84f70805201",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/4b5c3901d1a84f70805201"

 },

 {

 "Id": 5175,

 "AppCardId": "072ff6e3bdc6423dbfd985",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/072ff6e3bdc6423dbfd985"

 },

 {

 "Id": 5192,

 "AppCardId": "IF00\_USER05\_CP01\_X",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/IF00\_USER05\_CP01\_X"

 },

 {

 "Id": 5193,

 "AppCardId": "029abbf9e32a43b791734c",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/029abbf9e32a43b791734c"

 },

 {

 "Id": 5194,

 "AppCardId": "022e843f7b9742dd9214ad",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/022e843f7b9742dd9214ad"

 },

 {

 "Id": 5195,

 "AppCardId": "d429215ba147472786e9ca",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/d429215ba147472786e9ca"

 },

 {

 "Id": 5199,

 "AppCardId": "f5f3cb476758468980647d",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/f5f3cb476758468980647d"

 },

 {

 "Id": 5200,

 "AppCardId": "278aabeac4a24a40b0c845",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/278aabeac4a24a40b0c845"

 },

 {

 "Id": 5201,

 "AppCardId": "170e289e51254a9e89e617",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/170e289e51254a9e89e617"

 },

 {

 "Id": 5202,

 "AppCardId": "c55dcd653e144d8297def8",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/c55dcd653e144d8297def8"

 },

 {

 "Id": 5300,

 "AppCardId": "card555",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/card555"

 },

 {

 "Id": 5323,

 "AppCardId": "card55555",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00\_USER05/cards/card55555"

 }

 ],

 "Status": 1,

 "Company": null,

 "CountryCode": null,

 "KycStatus": 3,

 "OnBoardingDate": "2019-03-08T15:29:05"

 },

 {

 "Id": 877,

 "AppUserId": "IF00USER11",

 "Type": 1,

 "Role": -1,

 "Profile": {

 "Civility": 0,

 "FirstName": "Jean",

 "LastName": "IF00USER11",

 "MaidenName": null,

 "Birthdate": "1979-08-06T00:00:00",

 "Birthcity": "London",

 "BirthCountry": "GB",

 "Address": {

 "Street": "5 rue de la paix",

 "ZipCode": "75017",

 "City": "PAris",

 "Country": "FR"

 },

 "PhoneNumber": "07000000",

 "Email": "if00user11@gmail.com",

 "Alias": "IF00USER11",

 "Picture": {

 "Href": null

 },

 "EconomicActivity": 0,

 "Asset": null,

 "Income": null

 },

 "Credentials": null,

 "Amount": 600000,

 "AccountingBalanceAmount": 600000,

 "SubAccounts": [

 {

 "Id": 877,

 "AppAccountId": "IF00USER11",

 "Amount": 600000,

 "AccountingBalanceAmount": 600000,

 "DisplayName": "Jean IF00USER11",

 "CreationDate": "2019-03-26T17:24:54",

 "IsDefault": true,

 "Iban": "FR7816528001020AV000000OD78",

 "IbanHint": "FR78XXXXXXXXXXXXXXXXOD78"

 }

 ],

 "BankAccounts": [

 {

 "Id": 302,

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/bankaccounts/302"

 }

 ],

 "CBCards": [

 {

 "Id": 1050,

 "AppCardId": "IF00USER11\_CV01",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/IF00USER11\_CV01"

 },

 {

 "Id": 1565,

 "AppCardId": "YassineISCTa123",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/YassineISCTa123"

 },

 {

 "Id": 1566,

 "AppCardId": "YassineISCTa124",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/YassineISCTa124"

 },

 {

 "Id": 1567,

 "AppCardId": "YassineISCTa125",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/YassineISCTa125"

 },

 {

 "Id": 1568,

 "AppCardId": "CarteREFX1",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/CarteREFX1"

 },

 {

 "Id": 1569,

 "AppCardId": "CarteREFX2",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/CarteREFX2"

 },

 {

 "Id": 1570,

 "AppCardId": "CarteREFX3",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/CarteREFX3"

 },

 {

 "Id": 1571,

 "AppCardId": "CarteREFX4",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/CarteREFX4"

 },

 {

 "Id": 1572,

 "AppCardId": "CarteREFX5",

 "Href": "http://rest-integ.s-money.net/api/bib/api/users/IF00USER11/cards/CarteREFX5"

 }

 ],

 "Status": 1,

 "Company": null,

 "CountryCode": null,

 "KycStatus": 3,

 "OnBoardingDate": "2019-03-26T17:24:54"

 }

 ],

 "links": [

 {

 "url": "?page=1&perpage=5&pageindex=0&direction=next",

 "relation": "prev"

 },

 {

 "url": "?page=3&perpage=5&pageindex=877&direction=next",

 "relation": "next"

 }

 ]

}

**Conditions for displaying the previous / next link**

* If users count Equal to 1(one) then pagination system will not be displayed.
* If number of users who matches is equal to page size, link to previous page will be created if page number is greater than 1.
* If number of users who matches is greater than page size, link to next the page will be created. Link to previous page will be created unless page number is greater than 1.
* If number of users who matches is less than page size, link to previous page will be created unless page number is greater than 1.

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Opération non autorisée |
| 401 | 569 | Token oauth expiré |
| 401 | 570 | Token oauth invalide |

# Retrieve a user

**Request:**

GET /api/v[1.x]/users/{appuserid} HTTP/1.1

Accept: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "Id":114,

 "AppUserId":"client-112",

 "Role":1,

 "Type":1,

 "Profile":

 {

 “Civility”:0,

 "FirstName":"Jean",

 "LastName":"Dupont",

 "Birthdate":"1985-09-29T00:00:00",

 "Address":

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "Phonenumber":"0600000001",

 "Email":"jeandupont@s-money.fr",

"Alias":"jean-dupont"

 },

 "Amount":0,

"AccountingBalanceAmout":0,

 "SubAccounts":

[{

 "Id":"225",

 "AppAccountId":"87",

 "Amount":0",

"AccountingBalanceAmout":0,

"DisplayName":"Sub Account",

"IsDefault":true,

"Iban": "FR1420041010050500013M02606"

}, {…}],

 "BankAccounts":

[{

 "Id":2,

 "Href":"/api/users/{AppUserId}/BankAccounts/2"

}, {…}],

"cbcards":

[{

 "Id":12,

 "AppCardId": "card1",

 "Href":"/api/users/{AppUserId}/cbcards/card1

}, {…}],

 "Status":1

}

**Error cases :**

| **HTTP code**  | **Error code**  | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 404 | 147 | User cannot be found |

# Retrieve a user’S BALANCE

GET /api[/V1.1/users/eaqbu48da/balance](https://qa-api.s-money.net/api/V1.1/users/eaqbu48da/balance)

This endpoint allows partners to retrieve the user’s balance.

**Request**

GET /api[/V1.1/users/eaqbu48da/balance](https://qa-api.s-money.net/api/V1.1/users/eaqbu48da/balance)

**Response**

{

"AppUserId": "eaqbu48da",

"Amount": 0,

"DebtBalance": 100,

"AccountingBalanceAmount": 0

}

# Modify a user

**Request with strong authentication (MANDATORY & ONLY FOR AGENT PARTNERS ONCE THE USER RECORD HAS BEEN VALIDATED):**

PUT/api/sca/v1.1/users/{userid}/

This endpoint must be used for modifying User’s information once the User record has been validated.

The body is the same as the request without strong User authentication below

**Request without strong authentication :**

This endpoint must be used for modifying User’s information if the User record is validated.

PUT /api/v[1.x]/users/{appUserId}

Content-Type: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

{

 "Profile":

 {

"Civility":0, (optional)

 "FirstName":"Jean", (optional)

 "LastName":"Dupont", (optional)

 "Birthdate":"1985-09-29T00:00:00", (optional)

 "BirthCountry":"FR", (optional)

 "BirthCity":"Paris", (optional)BirthCountry":"FR", (optional)

 "BirthCity":"Paris", (optional)

 "Address": (optional)

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "Phonenumber":"0600000001", (optional)

 "Email":"jeandupont@s-money.fr" (optional)

 }

}

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "Id":114,

 "AppUserId":"client-112",

 "Role":1,

 "Type":1,

 "Profile":

 {

 "Civility":0,

 "FirstName":"Jean",

 "LastName":"Dupont",

 "Birthdate":"1985-09-29T00:00:00",

 "BirthCountry":"FR",

 "BirthCity":"Paris",

 "Address":

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "Phonenumber":"0600000001",

 "Email":"jeandupont@s-money.fr",

"Alias":"jean-dupont"

 },

 "Amount":0,

"AccountingBalanceAmout":0,

 "SubAccounts":

[{

 "Id":"225",

 "AppAccountId":"87",

 "Amount":0",

"AccountingBalanceAmout":0,

"DisplayName":"Sub Account",

"IsDefault":true,

"Iban": "FR1420041010050500013M02606"

}, {…}],

 "BankAccounts":

[{

 "Id":2,

 "Href":"/api/users/{AppUserId}/BankAccounts/2"

}, {…}],

"cbcards":

[{

 "Id":12,

 "AppCardId": "card1",

 "Href":"/api/users/{AppUserId}/cbcards/card1

}, {…}],

 "Status":1,

}

**If user’s kyc is validated, only following informations could be modified**

* Civility
* FirstName
* LastName
* Address
* Phonenumber
* Email

**Error cases :**

| **HTTP code**  | **Error code**  | **Description**  |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 400 | 704 | Request parameter not provided |
| 400 | 177 | Request parameter invalid |
| 404 | 147 | User cannot be found |
| 400 | 567 | Date of birth invalid |
| 400 | 377 | The user must be over 18 years old |
| 400 | 550 | Incorrect or already taken e-mail |
| 400 | 514 | Incorrect or already taken account Alias |
| 400 | 515 | Incorrect or already taken telephone number |

# Close the user account

**WARNING : This operation is irreversible. No action will enable to re-open the account, & the personnal data are deleted and no more reachable (GDPR). The only possible action is to open again a new account with the same informations.**

**Request:**

**PUT /api/v[1.x]/users/{appUserId}**

Accept: application/vnd.s-money.v1+json

Content-Type: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

{

 "Status":5 (mandatory)

}

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "Id":114,

 "AppUserId":"client-112",

 "Role":1,

 "Type":1,

 "Profile":

 {

 "Civility":0,

 "FirstName":"Jean",

 "LastName":"Dupont",

 "Birthdate":"1985-09-29T00:00:00",

 "Address":

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "Phonenumber":"0600000001",

 "Email":"jeandupont@s-money.fr",

"Alias":"jean-dupont"

 },

 "Amount":0,

"AccountingBalanceAmout":0,

 "SubAccounts":

[{

 "Id":"225",

 "AppAccountId":"87",

 "Amount":0",

"AccountingBalanceAmout":0,

"DisplayName":"Sub Account",

"IsDefault":true,

"Iban": "FR1420041010050500013M02606"

}, {…}],

 "BankAccounts":

[{

 "Id":2,

 "Href":"/api/users/{AppUserId}/BankAccounts/2"

}, {…}],

"cbcards":

[{

 "Id":12,

 "AppCardId": "card1",

 "Href":"/api/users/{AppUserId}/cbcards/card1

}, {…}],

 "Status":5

}

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token oAuth expired |
| 401 | 570 | Token oAuth invalid |
| 400 | 704 | Request parameter missing or not provided |
| 400 | 177 | Request parameter invalid |
| 404 | 147 | User cannot be found |
| 400 | 302 | BIC / IBAN unknown / not filled in, cannot transfer the remaining funds to the user's account |
| 400 | 703 | Unit transaction limit reached, unable to transfer the remaining funds to the user's account |
| 400 | 344 | Annual transaction limit reached, unable to transfer the remaining funds to the user's account |

# SEND USER’S ACCEPTED GCU

**Request with strong authentication (MANDATORY & ONLY FOR AGENT PARTNERS)**

POST /api/sca/v2.0/users/{AppUserId}/cgu

POST /api/sca/v1.1/users/{AppUserId}/cgu

{

 "cardCgu": "true",

 "partnerCgu": "true",

 "accountCgu": "true"

}

**Response :**

200 OK

 {

 "card": true,

 "partner": true,

 "account": true

}

| **Property**  | **Type**  | **Format** | **Example**  | **Description**  |
| --- | --- | --- | --- | --- |
| AppUserId | String | X(9) | Client-112 | User identifier in the third-party application. |
| cardCgu | String |  |  | S-money card cguexpected value: true |
| partnerCgu | String |  |  | partner cguexpected value: true |
| accountCgu | String |  |  | S-money account cguexpected value: true |

# Search for a user

**Request :**

GET parameters :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Property | Type  | Format | Example  | Default value  |
| page | Int | N(10) | Page number (starts with 1) | 1 |
| per\_page | Int | N(10) | Number of items per page | 50 |
| smoneyid | string | X(255) | List of S-money ID separated by commas |   |
| FirstName | string | X(140) | Name |   |
| LastName | string | X(140) | Surname |   |
| Email | string | X(255) |  |   |
| AppUserID | string | X(9) |  |   |
| Types | int | N(10) |   |   |
| PhoneNumber | String | X(100) |  |  |
| hasDebtBalance | Boolean |  | hasDebtBalance = true 🡺 return Users with debtBalance > 0 |  |
| kycStatus | int | N(10) | /V1.1/users?kycStatus=5 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**GET** **/api/v[1.x]/users?hasDebtBalance=true**

Authorization: Bearer ACCESS\_TOKEN

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "value":

[

 {

 "Id": 31065,

 "AppUserId": "TEST00001",

 "Type": 1,

 "Role": 2,

 "Profile": {

 "Civility": 1,

 "FirstName": "JS",

 "LastName": "JSU",

 "BirthName": null,

 "Birthdate": "1985-09-29T00:00:00",

 "Birthcity": "London",

 "BirthCountry": "GB",

 "BirthZipCode": null,

 "Address": {

 "Street": "2 rue des lilas",

 "ZipCode": "75001",

 "City": "Paris",

 "Country": "FR"

 },

 "PhoneNumber": "0033665091514",

 "Email": "jsjsu001@s-money.fr",

 "Alias": "alias",

 "Picture": {

 "Href": null

 },

 "EconomicActivity": 38,

 "Asset": null,

 "Income": null

 },

 "Credentials": null,

 "Amount": 12345,

 "DebtBalance": 151,

 "AccountingBalanceAmount": 139,

 "SubAccounts": [

 {

 "Id": 31065,

 "AppAccountId": "TEST00001",

 "Amount": 12345,

 "AccountingBalanceAmount": 139,

 "DisplayName": "JS JSU",

 "CreationDate": "2021-02-16T11:23:06",

 "IsDefault": true,

 "Iban": null,

 "IbanHint": null

 }

 ],

 "BankAccounts": null,

 "CBCards": null,

 "Status": 1,

 "Company": null,

 "CountryCode": null,

 "KycStatus": 6,

 "OnBoardingDate": "2021-02-16T11:23:06"

 }

 ],

 "links": null

}

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |

# block/UNBLOCK on a user (REVERSIBLE)

**Request :**

**PUT /api/v[1.x]/users/{appUserId}**

Content-Type: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

{

 "Status":7 (mandatory)

}

**NB : Put “Status”:1 to Unblock**

**Response :**

HTTP/1.1 200 OK

Content-Type: application/vnd.s-money.v1+json

{

 "Id":114,

 "AppUserId":"client-112",

 "Role":1,

 "Type":1,

 "Profile":

 {

 "Civility":0,

 "FirstName":"Jean",

 "LastName":"Dupont",

 "Birthdate":"1985-09-29T00:00:00",

 "Address":

 {

 "Street":"2 rue des chataîgners",

 "ZipCode":"75001",

 "City":"Paris",

 "Country":"FR"

 },

 "Phonenumber":"0600000001",

 "Email":"jeandupont@s-money.fr",

"Alias":"jean-dupont"

 },

 "Amount":0,

"AccountingBalanceAmout":0,

 "SubAccounts":

[{

 "Id":"225",

 "AppAccountId":"87",

 "Amount":0",

"AccountingBalanceAmout":0,

"DisplayName":"Sub Account",

"IsDefault":true,

"Iban": "FR1420041010050500013M02606"

 }, {…}],

 "BankAccounts":

[{

 "Id":2,

 "Href":"/api/users/{AppUserId}/BankAccounts/2"

}, {…}],

"cbcards":

[{

 "Id":12,

 "AppCardId": "card1",

 "Href":"/api/users/{AppUserId}/cbcards/card1

}, {…}],

 **"Status":7 (or "Status":1 if unblock)**

}

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 401 | 704 | Request parameter not provided |
| 401 | 177 | Request parameter invalid |
| 401 | 147 | User cannot be found |

# Get a user’s account statement

**Request:**

GET parameters :

| **Property**  | **Type**  | **Format** | **Example**  | **Default value**  |
| --- | --- | --- | --- | --- |
| Year | Int | N(4) |  | 2018 |
| Month | Int | N(2) |  | 1 |

**GET /api/v[1.x]/users/{appuserid}/statement?year=2018&month=1**

Content-Type: application/pdf

Accept: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

If successful, the above call returns an HTTP code 200, and returns the generated PDF.

**Response:**

HTTP/1.1 200 OK

Content-Type: application/pdf

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 401 | 704 | Request parameter not provided |
| 401 | 177 | Request parameter invalid |
| 401 | 147 | User cannot be found |
| 401 | 1100 | Year parameter invalid |
| 401 | 1101 | Month parameter invalid |

# Account statements callback

| **Paramètre** | **Type** | **Description** |
| --- | --- | --- |
| year | long | Year of the generated user statements |
| month | string | Month of the generated user statements |
| filename | string | Filename of the archive containing all the generated user statements |
| type | Int | 28 (UserStatementsReady) |

Post {"year":"2018","month":"10", “filename”:”2018\_10\_filename.zip“, "type":"24"}

# Get Account Statements archive

**Request:**

GET parameters :

| **Property**  | **Type**  | **Format** | **Example**  | **Default value**  |
| --- | --- | --- | --- | --- |
| Year | Int | N(4) | 2018 | 2018 |
| Month | Int | N(2) | 1 | 1 |
| Filename | String | N(30) | 2018\_10\_filename.zip | N/A |

**GET /api/v[1.x]/users/{appuserid}/statement?year=2018&month=1&filename=filename.zip** HTTP/1.1

Content-Type: application/zip

Accept: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS\_TOKEN

If successful, the above call returns an HTTP code 200, and returns the archive to download, containing all the user statements for the selected month.

**Response:**

HTTP/1.1 200 OK

Content-Type: application/zip

**Error cases :**

| **Code HTTP** | **Error code** | **Description** |
| --- | --- | --- |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 401 | 704 | Request parameter not provided |
| 401 | 177 | Request parameter invalid |
| 401 | 1100 | Year parameter invalid |
| 401 | 1101 | Month parameter invalid |
| 401 | 1102 | Filename parameter invalid (not found) |

# Economic Activity Catalog

| **Code INSEE 2003**  | **LIBELLE INSEE (niveau 3)** |
| --- | --- |
| 11 | Agriculteurs sur petite exploitation  |
| 12 | Agriculteurs sur moyenne exploitation |
| 13 | Agriculteurs sur grande exploitation |
| 21 | Artisans |
| 22 | Commerçants et assimilés |
| 23 | Chefs d'entreprise de 10 salariés ou plus |
| 31 | Professions libérales |
| 33 | Cadres de la fonction publique |
| 34 | Professeurs, professions scientifiques |
| 35 | Professions de l'information, des arts et des spectacles |
| 37 | Cadres administratifs et commerciaux d'entreprise |
| 38 | Ingénieurs et cadres techniques d'entreprise |
| 42 | Professeurs des écoles, instituteurs et assimilés |
| 43 | Professions intermédiaires de la santé et du travail social |
| 44 | Clergé, religieux |
| 45 | Professions intermédiaires administratives de la fonction publique |
| 46 | Professions intermédiaires administratives et commerciales des entreprises |
| 47 | Techniciens |
| 48 | Contremaîtres, agents de maîtrise |
| 52 | Employés civils et agents de service de la fonction publique |
| 53 | Policiers et militaires |
| 54 | Employés administratifs d'entreprise |
| 55 | Employés de commerce |
| 56 | Personnels des services directs aux particuliers |
| 62 | Ouvriers qualifiés de type industriel |
| 63 | Ouvriers qualifiés de type artisanal |
| 64 | Chauffeurs |
| 65 | Ouvriers qualifiés de la manutention, du magasinage et du transport |
| 67 | Ouvriers non qualifiés de type industriel |
| 68 | Ouvriers non qualifiés de type artisanal |
| 69 | Ouvriers agricoles |
| 71 | Anciens agriculteurs exploitants |
| 72 | Anciens artisans, commerçants, chefs d'entreprise |
| 74 | Anciens cadres |
| 75 | Anciennes professions intermédiaires |
| 77 | Anciens employés |
| 78 | Anciens ouvriers |
| 81 | Chômeurs n'ayant jamais travaillé |
| 83 | Militaires du contingent |
| 84 | Elèves, étudiants |
| 85 | Personnes diverses sans activité professionnelle de moins de 60 ans (sauf retraités) |
| 86 | Personnes diverses sans activité professionnelle de 60 ans et plus (sauf retraités) |