|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **API-Cards** | | | |  |
|  | | | | |
|  |  |  |  | |
| **Version** | BiB-1.33 |  |  | |
| **Reference** | S-money-API-Cards | | | |

Content

[1. REVISIONs 3](#_Toc79761306)

[2. Resources API V1.0/ V1.1 6](#_Toc79761307)

[2.1.1 Card 6](#_Toc79761308)

[2.1.2 SubAccountRef 8](#_Toc79761309)

[2.1.3 Creationtype 9](#_Toc79761310)

[2.1.4 CardLimits 9](#_Toc79761311)

[3. Resources API V2.0 9](#_Toc79761312)

[3.1.1 Card 9](#_Toc79761313)

[3.1.2 status 12](#_Toc79761314)

[3.1.3 event Status 12](#_Toc79761315)

[3.1.4 holder 13](#_Toc79761316)

[4. Endpoints to request 13](#_Toc79761317)

[5. Card order 14](#_Toc79761318)

[5.1.1 IdCard order v1.0/ V1.1 14](#_Toc79761319)

[Request Card order v1.0/ v1.1 (physical and virtual card) 14](#_Toc79761320)

[Response Card order v1.0 (physical and virtual card) 14](#_Toc79761321)

[Error codes 15](#_Toc79761322)

[5.1.2 CREATE A Card v2.0 16](#_Toc79761323)

[Request Create a Card v2.0 with strong customer authentication (mandatory & only for Agent partners) 16](#_Toc79761324)

[Request Create a Card v2.0 (physical and virtual card) (mandatory & only for PECI partners) 16](#_Toc79761325)

[Response Create a Card v2.0 (physical card) 16](#_Toc79761326)

[Error codes 16](#_Toc79761327)

[6. Get ISSUED cards list for SPECIFIC USER V1.0/ V1.1 17](#_Toc79761328)

[6.1.1 Get ISSUED cards list for SPECIFIC USER V1.0/ V1.1 17](#_Toc79761329)

[Request Get issued card lists for specific user V1.0/ V1.1 17](#_Toc79761330)

[Response Get issued card lists for specific user V1.0/ V1.1 17](#_Toc79761331)

[Error codes 18](#_Toc79761332)

[6.1.2 Get All cards by holder V2.0 18](#_Toc79761333)

[Request Get all cards by holder V2.0 18](#_Toc79761334)

[Response Get all card by holder V2.0 19](#_Toc79761335)

[Error codes 20](#_Toc79761336)

[7. Get ISSUED cards list for all users 20](#_Toc79761337)

[7.1.1 Get cards list for All USERs V1.0/ V1.1 20](#_Toc79761338)

[Request Get issued card lists for all users V1.0/ V1.1 20](#_Toc79761339)

[Response Get issued card lists for all users V1.0/ V1.1 20](#_Toc79761340)

[Error codes 21](#_Toc79761341)

[7.1.2 Get cards V2.0 21](#_Toc79761342)

[Request Get cards V2.0 21](#_Toc79761343)

[Response Get cards V2.0 21](#_Toc79761344)

[Error codes 23](#_Toc79761345)

[7.1.3 Get a specific ISSUED card details 23](#_Toc79761346)

[Request Get a specific issued card details V1.0/ V1.1 23](#_Toc79761347)

[Response Get a specific issued card details V1.0/ V1.1 23](#_Toc79761348)

[Error codes 24](#_Toc79761349)

[7.1.4 Get card details V2.0 24](#_Toc79761350)

[Request Get card details V2.0 24](#_Toc79761351)

[Response Get card details V2.0 24](#_Toc79761352)

[Error codes 26](#_Toc79761353)

[8. Card Refab 26](#_Toc79761354)

[8.1.1 Card refab v1.0/ V1.1 26](#_Toc79761355)

[Request Card Refabrication v1.0/ v1.1 26](#_Toc79761356)

[Response Card Refabrication v1.0/ v1.1 27](#_Toc79761357)

[Error codes 28](#_Toc79761358)

[8.1.2 Card refab V2.0 28](#_Toc79761359)

[Request card refab v2.0 with strong customer authentication (mandatory & only for Agent partners) 28](#_Toc79761360)

[Request card refab v2.0 28](#_Toc79761361)

[Response Card Refab v2.0 29](#_Toc79761362)

[Error codes 29](#_Toc79761363)

[9. Callbacks : 29](#_Toc79761364)

[10. Card Display v1.1 30](#_Toc79761365)

[11. Card Display v2.0 33](#_Toc79761366)

[12. CALCUL HMAC : 33](#_Toc79761367)

[13. PIN DISPLAY : 35](#_Toc79761368)

[13.1.1 Value uased For the Sandbox : 35](#_Toc79761369)

[13.1.2 Request : 36](#_Toc79761370)

[13.1.3 Response : 37](#_Toc79761371)

[13.1.4 decryption of block pin (Method proposed) 38](#_Toc79761372)

[14. Card SelfCare 40](#_Toc79761373)

[14.1.1 Selfcare V1.0/ V1.1 40](#_Toc79761374)

[Request V1.0/ V1.1 40](#_Toc79761375)

[Response](#_Toc79761376)

[14.1.2 Selfcare V2.0 41](#_Toc79761377)

[Request V2.0 41](#_Toc79761378)

[Response V2.0 42](#_Toc79761379)

[15. Card CancelLation V2.0](#_Toc79761380)

[Request Cancels a card v2.0 43](#_Toc79761381)

[Response Cancels a card v2.0 43](#_Toc79761382)

[Error codes 43](#_Toc79761383)

[16. Card Opposition V2.0 43](#_Toc79761384)

[Request Opposes a card v2.0 43](#_Toc79761385)

[Response Opposes a card v2.0](#_Toc79761386)

[Error codes 44](#_Toc79761387)

[17. CVX2 DISPLAY V2.0 44](#_Toc79761388)

[18. PAN DISPLAY V2.0 46](#_Toc79761389)

Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.Erreur ! Référence de lien hypertexte non valide.Erreur ! Signet non défini.**

# REVISIONs

|  |  |  |
| --- | --- | --- |
| **Version / revision** | **Author** | **Description** |
| BiB-v1.1 |  | First version of the doc for BiB |
| BiB-v1.2 | Mohamed Marouani | Add “ FintechCultureName” as mandatory input parameter |
| BiB-v1.3 | Farouk MERAWZA | Split the get cards list into two endpoints (issued and registered) |
| BIB-v1.4 | Simon ROUX | * Deleted obsolete § “ 4. GET CARDS LIST FOR A SPECIFIC USER” * Put “’Fintech culture code” into the Resources § * Rename §15 “Card Internet Payment Unblock” (instead of Card Foreign Payment Unblock” |
| BIB-v1.5 | Farouk MERAWZA | Added description for hmac/wishpin endpoint |
| BIB-v1.6 | Nabil HAMZA | Add useRandomPin field |
| BIB-v1.7 | Simon ROUX | Add “IsAlphaTest” as a new attribute for Card resource |
| BIB-v1.8 | Nabil HAMZA | Add “Get cards list for a specific user” API. |
| BIB-v1.9 | Giampiero MICCOLI | Add Unique Id and Bank Id into Card section §2.1 |
| BIB-v1.10 | Giampiero MICCOLI | Add PIN display section |
| BIB-v1.11 | Hadrien LECOQ | Added precision for hmac/wishpin |
| BIB-V1.12 | Hadrien LECOQ | Add ChannelCode description |
| BIB-V1.13 | Giampiero MICCOLI | Update UniqueId example result values |
| BIB-V1.14 | Marouani Mohamed | Update Pin Display Response struct with Example |
| BIB-V1.15 | Hadrien Lecoq | Add Failed Status |
| BIB-V1.16 | Marouani Mohamed | Algorithme de déchiffrement du PIN Block |
| BIB-V1.17 | Toan PHAM | CARD ORDER: error code 710 à 717 (CardId already exists) |
| BIB-V1.18 | Sébastien Bik | Update of all items. |
| BIB-V1.19 | Simon Roux | Clean up (examples, precision in description) + little corrections |
| BIB-V1.20 | Fahd Aloui | Card SelfCare |
| BIB-V1.20 | Sébastien Bik | Update of Card refabrication. Precision of the value for the selfcare fields. |
| BIB-V1.21 | Sébastien Bik | Update with the API V2.0 :  Card Order  Card Refab  [Get ISSUED cards list for SPECIFIC USER](#_Toc49420985)  [Get ISSUED cards list for all users](#_Toc49420986)  [Get a specific ISSUED card details](#_Toc49420987) |
| BIB-V1.21 | Sébastien Bik | Update on the substance of document.  Distinction between the various APIs versions |
| BIB-V1.22 | Sébastien Bik | Update on the substance of document.  Distinction between the various APIs versions |
| BIB-V1.23 | Sébastien Bik | Precision on the format of the various keys used on the cryptographic calls.  Integration of the Cancellation API.  Integration of the Opposition API. |
| BIB-V1.24 | Sébastien Bik | Pin display update.  Update of all Sandbox Urls. |
| BIB-V1.25 | Sébastien Bik | Updating the values used and received for the Get Pin.  New explanations on the modus operandi. |
| BIB-V1.26 | Sébastien Bik | Precision on the certificate format used for the Get Pin.  Precisions of the Mode and the padding of the 3DES used for encypher the KTA. |
| BIB-V1.27 | Sébastien Bik | URL update of the swagger available for the SandBox |
| BIB-V1.28 | Kawtar Tamazi  Sébastien Bik | Proofreading |
| BIB-V1.29 | Sébastien Bik | Used format of the KTA key in the encyphered process of the get PIN |
| BIB-V1.30 | Sébastien Bik | -Addition of the possibility of creating cards with deactivated contactless on an offer allowing it (API: v1.0/v1.1 and v2.0).  -Addition of the payment activity allowed by a client offer on the various Get, available only for v2.0 API. |
| BIB-V1.31 | Sébastien Bik | Clarification of cancellationReasonCode and oppositionReasonCode values on theAPI v2.0 |
| BIB-V1.32 | Sébastien Bik | -New description of the object : status in v2.0 API version.  -New label for pinCode (pinBlock) in the pindisplay API.  -New Enpoint for strong customer authentication for card creation and refabrication.  -Modification of the cancelation API signature without body. |
| BIB-V1.33 | Sébastien Bik | -decommissioning of the endpoint v1.1 to create virtual card  -use of the endpoint v2.0 to create virtual card  -New endpoints for Get Cvx2 in version 2.0  -New endpoint for Get Pan in version 2.0  -deletion in the documentation of the endpoints : creation of virtual card in stock mode and the endpoint associate an holder to a virtual card |

# Resources API V1.0/ V1.1

## Card

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| Id | Long | N(19) | Always "0" | Do not use.  The Card ID is used for an **internal usage** |
| AppCardId | String | X(255) | XPOLLENS-CP1 | It is the card’s external reference (Third party card ID) |
| IsAlphaTest | Int |  | 1 | This is related to an internal configuration. Alpha-tests cards are for NPS-Manufacturer chain validation:  1 = This card is used for alpha-test  0 = This car is not used for alpha-test |
| IsContactlessBlocked |  |  | 0 | 0= IsContactlessBlocked is false available on a client offer that allow it. 1 = IsContactlessBlocked is true available on a client offer that allow it. Null= Always Null for virtual card |
| Network | int | N(10) | 3 | Card’s network (3 = VISA) |
| Hint | String | X(16) | 1234XXXXXXXX5678 | Hint of the PAN |
| Name | String | X(100) | "Carte bancaire" | Card’s display name |
| Country | String | X(2) | "FR" | Card’s country (ISO 3166) |
| Type | int | N(10) | 1 | Card’s type:  1 = Classic virtual  2 = Classic physical  3 = Premium virtual  4 = Premium physical |
| ExpiryDate | DateTime | MM-YYYY | "10-2023" | Card’s expiration date |
| Status | int | N(10) | 1 | Card status:  0 = ordered  1 = sent  2 = activated  3 = expired  4 = opposed  5 = failed  6= deactivated  7=Canceled |
| OpposedReason | int | N(10) | 41 | It’s “null” when the card is not opposed. Otherwise filled:  7 = special conditions  34= suspected Fraud  41 = lost  43 = stolen  51 = accounts not stocked |
| AccountId | SubAccountRef | X(255) | Please check the next table « SUBACCOUTREF » for the more details | Associated subaccount (main account if not specified) |
| CreationType | CreationType |  | Please check the table « CREATIONTYPE » for the more details | Card’s creation type |
| FintechCultureName | String | X(256) | "FRENCH" | Do not use (obsolete).  It is the Fintech Culture Name |
| CardLimits | Limits object |  | Please check the table « CARDLIMITS» for more details | Card’s specific limits |
| Blocked | int | N(10) | 0 | Card’s temporary blocking, if it’s equal to :  0= The card is not temporary blocked  1= The card is temporary blocked |
| ForeignPaymentBlocked | Int |  | 1 | 0= The foreign payment is not temporary blocked  1= The foreign Payment is not temporary blocked |
| InternetPaymentBlocked | Int |  | 1 | 0= The internet payment is not temporary blocked  1= The internet payment is temporary blocked |
| UseRandomPin | Int |  | 1 | It’s refers to the PIN mode used:  0= WishPin (The holder chooses his own PIN)  1= RandomPin (The PIN is choosen by the bank) |
| UniqueId | String |  | 1652800001LEQrxSJ6NE2ydHmpBBE5zg | Bank Code +Branch Code+ InternalUniqueID |
| BankId | String |  | Always"30007 " | Natixis bank Id |
| CancellationReason | int |  | 0 | To precise the cancellation reason code:  1 : the customer no longer wants his card 2 : the customer left the bank 3 : deceased customer 4 : Partner request 6 : other reason (default value if not provided in the input paramete |

## SubAccountRef

Describes an account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| Id | Long | N(19) | Always "0" | Account ID/ Internal ID not provided |
| AppAccountId | String | X(255) | "xpollens1" | Thirdparty account ID |
| DisplayName | String | X(140) | "Jean Dupont" | Account display name (First name + Last name) |
| Iban | String | X(30) | Always "0" | Iban of the payment account |
| Alias | String | X(32) | Always "null" | User alias (User Nickmane) |
| PhoneNumber | String | X(100) | +33123456789 | Mobile phone number |
| Email | String | X(255) | jeandupont@s-money.fr | E-mail address |
| Href | url | X(255) | Always "null" | Details URL |

## Creationtype

Defines the creation mode/action for a card

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| RelatedCard | String |  | null | Related Card (Mandatory for a refabrication):  Not present and not used for creation = Action ‘0’ |
| Action | int | N(10) | 0 | Action code, Mandatory in the request  0 = Creation  2 = Refabrication |

## CardLimits

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| Transaction | int | N(10) | Always null | Transaction’s limit |
| ATMWeeklyAllowance | int | N(10) | 250 | ATM weekly Limit (Value in euro) |
| ATMWeeklyUsedAllowance | int | N(10) | 0 | Initialize to zero « 0 » (Value in euro) |
| MonthlyAllowance | int | N(10) | 2000 | Card payment Monthly limit (Value in euro) |
| MonthlyUsedAllowance | int | N(10) | 0 | Initialize to zero (Value in euro) |

# Resources API V2.0

## Card

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| offerPartnerCode | String |  | PartnerClassicPhysicalDebitVISA | The offer is provided by S-money, it refers to a card’s product for a specific partner and |
| partnerCode | String | X(50) | Partner1 | Partner’s code provided by S-money. |
| holderExternalRef | String |  | CustomerX | Partner’s holder reference |
| cardExternalRef | String | X(50) | CustomerXCard | Parner’s Card reference |
| expiryDate | String | X(7) | 01/2023 | Card expiry date |
| visualCodeSelected | String | X(4) | VSPV | Visual code selected between various values provided by S-money. |
| label | String | X(50) | MyCard1 | Label of the card (provided by the partner) |
| wishPin | bool | true/ false | false | Flag to specify if we use wishPIN or random PIN:  true=WishPin  false=RandomPin |
| isAlphaTest | bool | true/ false | true | Indicated if the card is an Alphatest card or not. This parameter is defined in the offer (by S-money) |
| isNfcActivated | Bool | True/false/ Null | true | 0= isNfcActivated is false available on a client offer that allow it. 1 = isNfcActivated is true available on a client offer that allow it. Null= Always Null for virtual card |
| status (cf.3.1.2) | String |  | SENT | **Card status:**  Ordered  Sent  Activated  Expired  Opposed  Failed  Deactivated  Canceled |
| creationDate | String |  | 2020-07-22T10:50:23.5938259 | The card creation date in S-money system |
| isBlocked | bool | true/ false | false | Indicates whether the card is block or not (If the card is blocked, no transaction is allowed) |
| globalLimitAtmSelected | Int(32) |  | 100 | It’s a card specific limit: the withdrawal limit on seven days slippery (The card is created with default limits defined in the offer, the value is in euros) |
| globalLimitPaymentSelected | Int(32) |  | 500 | It’s a card specific limits : the payment limit on Thirty slippery days (Value in euros) |
| uniqueId | String |  | 1652800001XJd7Mgyx-0a77qY86qpTOw | It’s the concatenation between (Bank code + Branch code +InternalUniqueID). And used to match the cardExternalRef and the wish pin chosen by the holder in the attribution wish pin process. |
| bankId | Int(32) |  | 30007 | Natixis bank Id |
| hint | string |  | 1234XXXXXXXX5678 | Hint of the Pan. |
| oldExternalRef | string |  |  | It’s the reference of the previous card in the refabrication process (Card mother). |
| isVadBlocked | bool | true/ false | true | true: the internet payment is temporary blocked  false: the internet payment is not temporary blocked |
| isGeoBlocked | bool | true/ false | false | true: the authorization will be refused if the country code of the transaction is different from the Agency country code. |
| OppositionReasoncode | string | Varchar(5) | lost | To precise the opposition reason code:  -41 = Lost  -34 = FraudSuspicon;  -43 = Stolen  -51 = (do not use)  -7 = Not received |
| channelCode | String |  | 66 | To precise the opposition channel:  04: desktop  66: mobile device  72: tablet |
| cancellationReasonCode | string | Varchar(25) | the customer left the bank | To precise the cancellation reason code:  CustomerNoLongerWantsCard = 1,  CustomerLeftBank = 2,  DeceasedCustomer = 3,  PartnerRequest = 4,  OtherReason = 6 |
| oppositionDeclarationDate | string | YYYY-MM-DD | 2020-11-24T22:11:10.973Z | Declaration date of the opposition by the holder |
| holder (cf.) | string |  |  | Holder’s information. |
| isPaymentAllowed | bool | true/ false | true | Indicate the type of card activity allow ( by the client offer) for physical present card transaction |
| isAtmWithdrawalAllowed | bool | true/ false | false | Indicate the type of card activity allow ( by the client offer) for physical present card transaction. It’ ll be false far an offer that don’t allow the withdrawal activity on the card. |
| isQuasiCashAllowed | bool | true/ false | true | Indicate the type of card activity allow ( by the client offer) for physical present card transaction |
| isWithdrawalAtTheCounterAllowed | bool | true/ false | true | Indicate the type of card activity allow ( by the client offer) for physical present card transaction |

## status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| cardStatusCode | Varchar(15) |  | SENT | Status of the card:  Ordered  Sent  Activated  Expired  Opposed  Failed  Deactivated  Canceled |
| description | Nvarchar(100) |  | sent | Status Description:  Ordered  Sent  Activated  Expired  Opposed  Failed  Deactivated  Canceled |
| eventStatus (cf.3.1.3) |  |  |  | Complement of the card Status.  Precision of the state of the status in asynchronous flow. |

## event Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| eventStatusCode |  |  |  | CREA (Demande de création) REFX (Demande de réfection sans changement de PAN) MAJR (Mise à jour des référentiels sans demande de personnalisation) OPPO (Demande de mise en opposition)  ANNL (Demande d’annulation) |
| Description |  |  |  | Demande de création (CREA) Demande de réfection sans changement de PAN (REFX) Mise à jour des référentiel sans demande de personnalisation (MAJR) Demande de mise en opposition (OPPO) Demande d’annulation (ANNL) |
| Status |  |  |  | PENDING  SUCCEEDED  FAILED  SENT  REJECTED  UNPROCESSABLE  LOADED |

## holder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| holderExternalRef | string | Varchar(50) | DEMO\_0002 | Partner’s holder reference |
| partnerCode | String | Varchar(50) | Partner1 | Partner’s code (Provided by S-money) |
| holderFirstName | String | Varchar(50) | JEAN | The first name of the holder |
| holderLastName | String | Varchar(50) | DUPONT | The last name of the holder |
| holderPhoneNumber | String | Varchar(16) | 0033752267114 | The phone number of the holder |
| holderEmail | String | Varchar(50) | Jean\_dupont@s-money.fr | The email of the holder |

# Endpoints to request

The various Endpoints used to send requests are:

* The API version V1.0 with a Bearer token
* The API version V1.1 with a STS token
* The API version V2.0 with a STS token

# Card order

## IdCard order v1.0/ V1.1

**Remark:** There is no swagger available for the card order V1.0/ V1.1.

### Request Card order v1.0/ v1.1 (physical and virtual card)

Request v1.0: (POST) /api/v1.0/users/{{appuserid}}/cards/

{

"AppCardId":"XPOLLENS-CP1", *(mandatory)*

"Type": 2, *(mandatory)*

"name": "CARTE PHYSIQUE CLASSIQUE M.JEAN DUPOND ", *(optional)*

"UseRandomPin": 1, *(optional)*

"isContactlessBlocked": 1, *(optional)*

"FintechCultureName": "FRENCH", *(optional, not used)*

    "AccountId":    *(optional, not used)*

     {

        "AppAccountId":"XPOLLENS1"

     },

   "CreationType":  {

        "Action": 0  *(mandatory, 0 for CREA)*

        }

}

### Response Card order v1.0 (physical and virtual card)

**Response v1.0 :** 200 OK

{

    "Id": 0, *(harcoded)*

    "AppCardId": "XPOLLENS-CP1",

    "IsAlphaTest": 0,

    "IsContactlessBlocked": 1,

    "Network": 3, *(hardcoded)*

    "Hint": "1234XXXXXXXX5678",

    "Name": "PHYSICAL CLASSIC CARD FOR M.JEAN DUPOND",

    "Country": "FR",

    "Type": 2,

    "ExpiryDate": "10-2023",

    "Status": 1,

    "OpposedReason": **null**,

    "AccountId": {

        "Id": 0, (*harcoded)*

        "AppAccountId": "xpollens1",

        "DisplayName": "jordan martin",

        "Iban": "0", (*harcoded)*

        "Alias": **null**, (*harcoded)*

        "PhoneNumber": "0033645799655",

        "Email": "Jean\_dupont@s-money.fr",

        "Href": **null** (*harcoded)*

    },

    "CreationType": {

        "RelatedCard": **null**,

        "Action": 0

    },

    "FintechCultureName": "French", (*harcoded)*

    "CardLimits": {

        "Transaction": **null**,

        "ATMWeeklyAllowance": 300,

        "ATMWeeklyUsedAllowance": 0, *(always 0 at creation)*

        "MonthlyAllowance": 3000,

        "MonthlyUsedAllowance": 0 *(always 0 at creation)*

    },

    "Blocked": 0,

    "ForeignPaymentBlocked": 0,

    "InternetPaymentBlocked": 0,

    "UseRandomPin": 1,

    "UniqueId": "1652811111MUaOC0Yntk2qXIsTKAMg0A",

    "BankId": "30007",

    "CancellationReason": 0

}

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 400 | 715 | Request parameter not provided or invalid |
| 400 | 364 | Invalid operation |
| 404 | 717 | CardId already exist |
| 404 | 147 | User cannot be found |
| 404 | 1069 | No virtual pan available for attribution |

## CREATE A Card v2.0

**Remark:** Swagger available on the sandbox environment: <https://sb-api.xpollens.com/swagger/index.html>

### Request Create a Card v2.0 with strong customer authentication (mandatory & only for Agent partners)

In order to perform the customer strong authentication you must use this endpoint :

(POST)/api/sca/v2.0/card/{{holder}}

The body is the same as therequest without strong customer authentication below.

### Request Create a Card v2.0 (physical and virtual card) (mandatory & only for PECI partners)

Request v2.0: (POST) /api/v2.0/card

{

"offerPartnerCode":"partnerExempleClassicPhysical", *(mandatory)*

"holderExternalRef": "ExampleClassicPhysical" , *(mandatory)*

"cardExternalRef": "Xpollens\_CP1", *(mandatory)*

"visualCodeSelected": "VXST", *(optional)*

"label": "PHYSICAL CLASSIC CARD FOR M.JEAN DUPOND", *(optional)*

"wishPin":true *(optional)*

"isNfcActivated":true *(optional)*

}

### Response Create a Card v2.0 (physical card and virtual card)

Response v2.0 : 201 OK

{

"cardExternalRef": "Xpollens\_CP1",

"partnerCode": "Partne1"

}

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 001 | Missing parameter |
| 400 | 002 | The partner offer does not exist |
| 400 | 003 | A card with the same ExternalRefCode exists |
| 400 | 004 | The holder does not exist |
| 400 | 007 | Error during pan computation |
| 400 | 010 | Label's character number should be lower than 50 |
| 400 | 018 | Partner not allowed to use this offer |



# Get ISSUED cards list for SPECIFIC USER V1.0/ V1.1

## Get ISSUED cards list for SPECIFIC USER V1.0/ V1.1

**Remark:** There’ s no swagger available for the car order V1.0/ V1.1

### Request Get issued card lists for specific user V1.0/ V1.1

**Request V1.0/ V1.1: (**GET) /api/v1.0/users/{appuserid}/cards/issued/

### Response Get issued card lists for specific user V1.0/ V1.1

[

{

        "Id": 0, *(harcoded)*

        "AppCardId": "Xpollens\_CP2",

        "IsAlphaTest": 0,

        "IsContactlessBlocked": 0,

        "Network": 3, *(harcoded)*

        "Hint": "1234XXXXXXXX5678",

        "Name": **null**,

        "Country": "FR",

        "Type": 2,

        "ExpiryDate": "06-2023",

        "Status": 2,

        "OpposedReason": **null**,

        "AccountId": {

            "Id": 0, *(harcoded)*

            "AppAccountId": "xpollens2",

            "DisplayName": "jordan martin",

            "Iban": "0", *(harcoded)*

            "Alias": **null**, *(harcoded)*

            "PhoneNumber": "0033645799655",

            "Email": "xpollens123@s-money.fr",

            "Href": **null** *(harcoded)*

        },

        "CreationType": **null**,

        "FintechCultureName": "French", *(harcoded)*

        "CardLimits": {

            "Transaction": **null**,

            "ATMWeeklyAllowance": 300, (300 euros)

            "ATMWeeklyUsedAllowance": 0.0,

            "MonthlyAllowance": 1000,

            "MonthlyUsedAllowance": 49.97 (49.97 euros)

        },

        "Blocked": 0,

        "ForeignPaymentBlocked": 0,

        "InternetPaymentBlocked": 0,

        "UseRandomPin": 1,

        "UniqueId": "1652811111/7Y5rLX9hUuPffVorh82rg",

        "BankId": "30007",

        "CancellationReason": 6

    }

]

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 401 | 570 | Token OAuth invalid |
| 400 | 715 | Request parameter not provided or invalid |
| 400 | 364 | Operation invalid |
| 404 | 710 | CardId already exist |
| 404 | 147 | User cannot be found |

## Get All cards by holder V2.0

Swagger available for the car order V2.0 on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

### Request Get all cards by holder V2.0

Request: Get /api/v2.0/card/holder/{holderExternalRef}

### Response Get all card by holder V2.0

Response v2.0 : 200 OK

[

{

"label": null,

"isAlphaTest": false,

"cardExternalRef": "Xpollens-CP3",

"status": {

"cardStatusCode": "SENT",

"description": "Sent",

"eventStatus": {

"eventStatusCode": "CREA",

"description": "Demande de création",

"status": "SUCCEEDED"

}

},

"creationDate": "2020-07-27T16:38:43.1058737",

"expiryDate": "06-2023",

"visualCodeSelected": "XVTU",

"isBlocked": false,

"globalLimitAtmSelected": 300,

"globalLimitPaymentSelected": 1000,

"uniqueId": "1652811111/7Y5rLX9hUuPffVorh82rg",

"bankId": 30007,

"hint": "9888XXXXXXXX7011",

"partnerCode": "Partne1",

"offerPartnerCode": « partnerExampleClassicVirual ",

"wishPin": false,

"oldExternalRef": null,

"isVadBlocked": false,

"isGeoBlocked": false,

"holderExternalRef": "xpollens1",

"oppositionReasonCode": null,

"cancellationReasonCode": "OtherReason"

"isNfcActivated": true

"isPaymentAllowed": true

"isAtmWithdrawalAllowed": true

"isQuasiCashlAllowed ": true

"isWithdrawalAtTheCounterAllowed ": true

}

]

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 004 | The holder does not exist |
| 400 | 014 | The partner does not exist |

# Get ISSUED cards list for all users

## Get cards list for All USERs V1.0/ V1.1

**Remark**: There’ s no swagger available for the car order V1.0/ V1.1

### Request Get issued card lists for all users V1.0/ V1.1

Request V1.0/ V1.1: (GET) /api/v1.0/cards/issued

### Response Get issued card lists for all users V1.0/ V1.1

Response : 200 OK

[

{

        "Id": 0, *(harcoded)*

        "AppCardId": "Xpollens\_CP4",

        "IsAlphaTest": 0,

        "IsContactlessBlocked": 0,

        "Network": 3, *(harcoded)*

        "Hint": "1234XXXXXXXX5678",

        "Name": **null**,

        "Country": "FR",

        "Type": 2,

        "ExpiryDate": "09-2023",

        "Status": 2,

        "OpposedReason": **null**,

        "AccountId": {

            "Id": 0, *(harcoded)*

            "AppAccountId": "xpollens2",

            "DisplayName": "jordan martin",

            "Iban": "0", *(harcoded)*

            "Alias": **null**, *(harcoded)*

            "PhoneNumber": "0033645799655",

            "Email": "xpollens123@s-money.fr",

            "Href": **null** *(harcoded)*

        },

        "CreationType": **null**,

        "FintechCultureName": "French", *(harcoded)*

        "CardLimits": {

            "Transaction": **null**,

            "ATMWeeklyAllowance": 300,

            "ATMWeeklyUsedAllowance": 0, (Always 0)

            "MonthlyAllowance": 1000,

            "MonthlyUsedAllowance": 0 (Always 0)

        },

        "Blocked": 0,

        "ForeignPaymentBlocked": 0,

        "InternetPaymentBlocked": 0,

        "UseRandomPin": 1,

        "UniqueId": "1652811111/7Y5rLX9hUuPffVorh82rg",

        "BankId": "30007",

        "CancellationReason": 6

    }

]

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 401 | 570 | Token OAuth invalid |
| 404 | 147 | User cannot be found |

## Get cards V2.0

Swagger available for the car order V2.0 on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

### Request Get cards V2.0

Request V2.0: (GET) /api/v2.0/card

### Response Get cards V2.0

Response v2.0 : 200 OK

[

{

"label": null,

"isAlphaTest": false,

"cardExternalRef": "Xpollens-CP5",

"status": {

"cardStatusCode": "SENT",

"description": "Sent",

"eventStatus": {

"eventStatusCode": "CREA",

"description": "Demande de création",

"status": "SUCCEEDED"

}

},

"creationDate": "2020-07-27T16:38:43.1058737",

"expiryDate": "06-2023",

"visualCodeSelected": "XVTU",

"isBlocked": false,

"globalLimitAtmSelected": 300,

"globalLimitPaymentSelected": 1000,

"uniqueId": "1652811111/7Y5rLX9hUuPffVorh82rg",

"bankId": 30007,

"hint": "9999XXXXXXXX7011",

"partnerCode": "Partne1",

"offerPartnerCode": "partnerExampleClassicPhysical",

"wishPin": false,

"oldExternalRef": null,

"isVadBlocked": false,

"isGeoBlocked": false,

"holderExternalRef": "xpollens2",

"oppositionReasonCode": null,

"cancellationReasonCode": "OtherReason"

"isNfcActivated": true

"isPaymentAllowed": true

"isAtmWithdrawalAllowed": true

"isQuasiCashlAllowed ": true

"isWithdrawalAtTheCounterAllowed ": true

}

]

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 014 | The partner does not exist |
| 400 | 018 | Partner not allowed to use this offer |

## Get a specific ISSUED card details

**Remark:** There’ s no swagger available for the car order V1.0/ V1.1

### Request Get a specific issued card details V1.0/ V1.1

Request : (GET) /api/v1.0/users/{appuserid}/cards/issued/{appcardid}

### Response Get a specific issued card details V1.0/ V1.1

**Response :** 200 OK

[

{

        "Id": 0, *(harcoded)*

        "AppCardId": "Xpollens\_CP7",

        "IsAlphaTest": 0,

        "IsContactlessBlocked": 0,

        "Network": 3, *(harcoded)*

        "Hint": "1111XXXXXXXX5678",

        "Name": **null**,

        "Country": "FR",

        "Type": 2,

        "ExpiryDate": "10-2023",

        "Status": 2,

        "OpposedReason": **null**,

        "AccountId": {

            "Id": 0, *(harcoded)*

            "AppAccountId": "xpollens2",

            "DisplayName": "jordan martin",

            "Iban": "0", *(harcoded)*

            "Alias": **null**, *(harcoded)*

            "PhoneNumber": "0033645799655",

            "Email": "xpollens123@s-money.fr",

            "Href": **null** *(harcoded)*

        },

        "CreationType": **null**,

        "FintechCultureName": "French", *(harcoded)*

        "CardLimits": {

            "Transaction": **null**,

            "ATMWeeklyAllowance": 300, (300 euros)

            "ATMWeeklyUsedAllowance": 0.0,

            "MonthlyAllowance": 1000,

            "MonthlyUsedAllowance": 22.97 (22.97 euros)

        },

        "Blocked": 0,

        "ForeignPaymentBlocked": 0,

        "InternetPaymentBlocked": 0,

        "UseRandomPin": 1,

        "UniqueId": "1652811111/7Y5rLX9hUuPffVorh22rg",

        "BankId": "30007",

        "CancellationReason": 6

    }

]

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 401 | 570 | Token OAuth invalid |
| 404 | 147 | User cannot be found |
| 400 | 195 | Card not found |

## Get card details V2.0

swagger available for the car order V2.0 on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

### Request Get card details V2.0

**Request V2.0 :** GET /api/v2.0/card/{cardExternalRef}

### Response Get card details V2.0

Response v2.0 : 200 OK

{

    "label": " Carte Example Classique virtuelle",

    "isAlphaTest": **false**,

    "cardExternalRef": "XPOLLENS1\_C9",

    "status": {

"cardStatusCode": "SENT",

"description": "Sent",

"eventStatus": {

"eventStatusCode": "CREA",

"description": "Demande de création",

"status": "SUCCEEDED"

}

},

    "oppositionReasonCode": "",

    "creationDate": "2020-08-12T09:14:41.0770573",

    "expiryDate": "09-2023",

    "visualCodeSelected": "UVWX",

    "isBlocked": **false**,

    "globalLimitAtmSelected": 300,

    "globalLimitPaymentSelected": 1000,

    "holder": {

        "holderExternalRef": "xpollens1",

        "partnerCode": "oney",

        "holderFirstName": "jordan",

        "holderLastName": "martin",

        "holderEmail": "xpollens123@s-money.fr",

        "holderPhoneNumber": "0645799655"

    },

    "offerPartner": {

        "offerPartnerCode": " partnerExampleClassicPhysical ",

        "partnerCode": "Partner1"

    },

    "uniqueId": "1652811111+NJL0UirckOqr8M8u9S/Yg",

    "bankId": 30007,

    "hint": "2222XXXXXXXX6238",

    "isGeoblocked": **false**,

    "isVadBlocked": **false**,

    "wishPin": **false**,

    "oldExternalRef": **null**,

    "cancellationReasonCode": "OtherReason"

"isNfcActivated": true

"isPaymentAllowed": true

"isAtmWithdrawalAllowed": true

"isQuasiCashlAllowed ": true

"isWithdrawalAtTheCounterAllowed ": true

}

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 006 | The card does not exist |
| 400 | 014 | The partner does not exist |

# Card Refab

The card can be remanufactured only in case of malfunction. If it’s not the case (Ex: If the customer didn’t receive his card, then he must create a new card by opposing the one not received).

PS: You can only remanufacture the last card issued (RelatedCard/ oldCardExternalRef) and not the previous ones (Already remanufactured).

## Card refab v1.0/ V1.1

There’ s no swagger available for the car order V1.0/ V1.1

### Request Card Refabrication v1.0/ v1.1

Request: (POST) /api/v1.0/users/{appuserid}/cards/

{

"AppCardId":"XPOLLENS-CP1\_X", *(mandatory, appCardId of the refabricated card)*

"Type": 2, *(mandatory)*

"name": "REMAKE THE CARD FOR M.JEAN DUPOND ", *(optional)*

"UseRandomPin": 1, *(optional)*

"FintechCultureName": "FRENCH", *(optional, not used)*

    "AccountId":    *(optional, not used)*

     {

        "AppAccountId":"XPOLLENS1"

     },

   "CreationType":  {      *(mandatory)*

"RelatedCard":  *(mandatory for the refabrication)*

{"AppCardId": " XPOLLENS-CP1"}, *(mandatory mother’s card AppCardId)*

         "Action": 2  *(mandatory, 2 for refabrication)*

        }

}

### Response Card Refabrication v1.0/ v1.1

**Response V1.0 :**

200 OK

{

    "Id": 0, (*harcoding*)

    "AppCardId": "XPOLLENS-CP1\_X ",

    "IsAlphaTest": 0,

    "IsContactlessBlocked": 0,

    "Network": 3, (*harcoding*)

    "Hint": "1111XXXXXXXX3239", *(Same Hint as the mother’s card)*

    "Name": "REMAKE THE CARD FOR M.JEAN DUPOND ",

    "Country": "FR",

    "Type": 2,

    "ExpiryDate": "11-2023",

    "Status": 1,

    "OpposedReason": **null**,

    "AccountId": {

        "Id": 0, (*harcoding*)

        "AppAccountId": "XPOLLENS1",

        "DisplayName": "Jean Dupond", (user’s first/last name)

        "Iban": "0", (*harcoding*)

        "Alias": **null**, (*harcoding*)

        "PhoneNumber": "0673434543", (user’s phone number)

        "Email": "sandoney2@s-money.fr", (user’s Email)

        "Href": **null** (*harcoding*)

    },

    "CreationType": {

        "RelatedCard": {

            "Id": **null**,

            "AppCardId": "XPOLLENS-CP1" (*appCardId of mother’s card*)

        },

        "Action": 2 (2 means “refabrication”)

    },

    "FintechCultureName": "French", (*harcoding*)

    "CardLimits": {

        "Transaction": **null**, (*harcoding*)

        "ATMWeeklyAllowance": 300, *(AtmWeeklyAllowance of mother’s card)*

 "ATMWeeklyUsedAllowance": 0, (hardcoded to 0.Use “Get Card API” to get the actual

ATMWeeklyUsedAllowance corresponding to mother card)

        "MonthlyAllowance": 3000, *(AtmWeeklyAllowance of mother’s card)*

        "MonthlyUsedAllowance": 0 (hardcoded to 0. Use “Get Card API” to get the actual

monthlyUsedAllowance corresponding to mother card)

    },

    "Blocked": 0, *(Same value as the mother’s card)*

    "ForeignPaymentBlocked": 0, *(Same value as the mother’s card)*

    "InternetPaymentBlocked": 0, *(Same value as the mother’s card)*

    "UseRandomPin": 1,

    "UniqueId": "1652811111642cdjk6bUaoRsxbbQIrAg",

    "BankId": "30007",

    "CancellationReason": 0

}

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 401 | 362 | Operation not authorised |
| 401 | 569 | Token OAuth expired |
| 401 | 570 | Token OAuth invalid |
| 400 | 715 | Request parameter not provided or invalid |
| 400 | 364 | Operation invalid |
| 404 | 710 | CardId already exist |
| 404 | 147 | User cannot be found |
| 400 | 704 | Missing Parameter |

## Card refab V2.0

Swagger available on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

### Request card refab v2.0 with strong customer authentication (mandatory & only for Agent partners)

In order to perform the customer strong authentication you must use this endpoint :

(POST)/api/sca/v2.0/card/refabricate/{{holder}}

The body is the same as therequest without strong customer authentication below.

### Request card refab v2.0

POST/api/v2.0/card/refabricate

{

"CardExternalRef":"Xpollens\_CP2\_X", *(mandatory)/ (New partner’s card reference)*

"oldExternalRef": "Xpollens\_CP2", *(mandatory)/ (Card mother’s reference)*

"visualCodeSelected": "VXST", *(optional)*

"label": "Remake the card" *(optional)*

}

### Response Card Refab v2.0

Response v2.0 : 201 OK

{

"cardExternalRef": "Xpollens\_CP2\_X",

"partnerCode": "Partner1"

}

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 003 | A card with the same ExternalRefCode exists |
| 400 | 006 | The card does not exist |
| 400 | 014 | The partner does not exist |
| 500 | 008 | Unable to process the remanufacturing request because the status is invalid |
| 500 | 009 | This card cannot be refabricated |
| 500 | 017 | Due to the cancellation of the card, this transaction cannot be completed. |

# Callbacks :

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Type** | **Description** |
| id | long | CardID |
| reference | String | AppCardId |
| type | Int | Callbacks card = 21 |
| cardType | Int | 1 : classic virtual  2 : classic physical  3 : premium virtual  4 : premium physical |
| action | Int | 0 = Creation  1 = Cancellation // Not used yet  2 = Refabrication  3 = Renewal // Not used yet  4 = Update |
| status | Int | 0 = Ordered  1 = Sent  2 = Actived  3 = Expired  4 = Opposed  5= Failed/Rejected  6=Deactivated  7=Cancelled |
| opposedReason | Int | "0" : no opposition  "07" : special conditions "34" : Suspected fraud "41" : Lost "43" : Stolen "51": unfunded account |

Example:

"Payload": {

"id": 0,

"reference": "card-123",

"type": 21,

"cardType": 4,

"action": 0,

"status": 1,

"opposedReason": 0

},

# Card Display v1.1

This version of card display doesn’t work if you create a virtual card with the endpoint v2.0.

Used the new version of card display v2.0.

Request for the card’s PAN, Cvx2 and end validity date.

The values in the answer are encrypted in RSA using the public key given in the request.

**Request:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| RSAPublicKey | string | X(16) |  | Represents the RSA pem public key coded in base64 String. |

**Sample request:**

Request : (POST) /api/v1.0/cards/{appcardid}/display

{

    "RSAPublicKey": ""

}

**Response:**

The response is composed of an http status code, and data in the body.

In normal flow without errors, these are the expected response fields:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| TransactionIdentifier |  |  | "213514", |  |
| cardIDToken | string | X(16) | xpollens\_CP1 | Card Identifier (Card external Ref) |
| signKeyIndex | string | X(4) | Always « null » | FUTURE Usage  Index of the buffer encryption key. |
| encryptKeyIndex | string | X(3) | Always « null » | FUTUR Usage  Index of the signature key |
| signature | string | X(16) | Always « null » | FUTUR Usage |
| buffer | object |  |  | **[<Buffer\_to\_encrypt>](#_<Buffer_à_chiffrer>_:_message)** **crypted using the RSA algorithm** with a public key passed in the field **{RSAPublicKey }**. |
| IsSuccess | boolean |  | true | Indicates if the response is an error or success |

<Buffer\_to\_encrypt> : json message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| alea | string | X(16) | qD1ey9Z9p0X2X2cV | Alea (can be ignored) |
| primaryAccountNumber | string | X(16) | 1234042345678901 | PAN |
| cardExpirationDate | string | X(4) | 1023 | Card Expiration date |
| cvv2 | string | X(3) | 506 | Cvv2 |

**Sample WS response (the body):**

HTTP/1.1 200 OK

{

    "TransactionIdentifier": "213514",

    "cardIDToken": "               xpollens\_CP1",

    "signKeyIndex": **null**,

    "encryptKeyIndex": **null**,

    "signature": **null**,

    "buffer": {

        "encryptedData": "dftjonX4QPmWyrOIFSYu1RQ62E33UNashGs89n7fYV1RbYKSbCNOMGPdzFN8J3Gh//YUCLvOAPmDzCPA/gDSZ59HltHSgLVN/SziSuY8teNcpJhRZC4GFrACrgcZVqwEelBsEPaSyVfog5qvym3tKKYcNs2atPDZJEoO1HRu42CzNuyTUby6+eVssrjdtVjYoY+ds5YqXdSGvwvnKRpG8gwSvgmWyS+saKz7lDXXLlhF7Iw0TA1qYEBvdvVpy3MCqHOgVrQuWDYnJX/99gOLSiWgM14gCJi67Q1jfjlPoDnpZIZp4TH2Wz5Z6ALwm0HMp2Ena0NK+YLGH4y27FKD9A=="

    },

    "IsSuccess": **true**

}

Encrypted buffer content (json):

{"alea":"qD1ey9Z9p0X2X2cV","primaryAccountNumber":"1234042345678901","cardExpirationDate":"1023","cvv2":"506"}

# CALCUL HMAC :

The Hmac encrypted request is necessary for the PIN definition (wishpin).

In the request, the hmac must contains 3 data : bankId, UniqueId, TimeStamp. The response will contain the encrypted hmac.

The encrypted hmac and the used timestamp are mandatory data for the

**Request :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| RSAPublicKey | string | X(16) |  | Represents the RSA pem public key coded in base64 String. |
| HmacData | string | X(2732) | SXQBUAAAERf//w== | Binary data buffer encoded in base 64.  Hmac must contains = bankId, UniqueID, TimeStamp |

**Sample request:**

POST /api/v1.0/cards/{appcardid}/hmac

{

    "RSAPublicKey": "",

    "HmacData":"SXQBUAAAERf//w=="

}

**Response :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| TransactionIdentifier |  |  | 213514 |  |
| cardIDToken | string | X(16) | Xpollenx\_CP2 | Card Identifier |
| signKeyIndex | string | X(4) | Always « null » | FUTURE Usage  Index of the buffer encryption key. |
| encryptKeyIndex | string | X(3) | Always « null » | FUTUR Usage  Index of the signature key |
| signature | string | X(16) | Always « null » | FUTUR Usage |
| buffer | object |  |  | **[<Buffer\_to\_encrypt>](#_<Buffer_à_chiffrer>_:_message)** **crypted using the RSA algorithm** with a public key passed in the field **{RSAPublicKey }**. |
| IsSuccess | boolean |  | true | Indicates if the response is an error or success |

* <Buffer\_réponse\_calcul\_HMAC> : message au format json

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| encryptedData | string |  | nJ2vucq7e24hStMpDVzcXR4tqxeITht5FnrTNYq/c6+Cch7ADohB6BhtoEoUycybkfMSqnXAi8k1WN6RZ7wIHOIjAzGoafQoX0PPTbBq8F0QB8Qen/Ba3/YbkcewX/quuDowvKlJpMadTqFaDV+lm0jYaAh18CEZ6aw3SM2il9ODWrEGpJd5U8oMIHMthJwc4guHmsxEG1G3byJ5ZT9rBWgSQz4EnFS7KgjTvmZjkhtJhPWPDPVV0PGWgw2T04iLSDUiIMLmQGXzOYp4uMoj0PgFzAQ5ezmpbvo58Ej3MneaNb78t8I+10qvxFReZAXcfp4ToX6oFuYbEip0v6h7Bw== | A string of characters - encrypted JSON message containing sensitive HMAC response data <Données\_réponse\_calcul\_HMAC>: json-sized message - the "alea" field encrypted by the RSA algorithm with the public key contained in the 'rsaPemPublicKey' field received in the input WS message. |

* <Données\_réponse\_calcul\_HMAC> : message au format json

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| hmac | string | X(16) | 4561042345678901 | Value of HMAC encoded in base 64 |
| returnCodeMetier | string | X(4) | 0219 | Expiration date |

**Sample WS response (the body):**

HTTP/1.1 200 OK

{

    "TransactionIdentifier": "213514",

    "cardIDToken": "               Xpollenx\_CP2",

    "signKeyIndex": **null**,

    "encryptKeyIndex": **null**,

    "signature": **null**,

    "buffer": {

        "encryptedData": "LeZo4q+uQdev9VYM76pq1Aqz5h2cdk2m2bibfsrivxn7PX0cS2Z3B8ul4pbu1rG3L4wKqZbX+5tih+o6DqbI9MtTsbP6q6Z24s0FOX8UU2qWzaMK5bF4NNME0oFP/HYmc623hD7h3tQVTJZydc6Ubqreu+4GhT1Fi14Pe5wqwWvOXVuOavb5OrOAUl3tn6A34lIaF8GsEj3OBTeZdQP2zyzeTz3K0CXG7O5LD3Txhz5rLPUVi1r0Vrjw31cYcy0N2bJkxFsYXgxyC6DKOPEYsHFi1rmBrM+pfe1Ah0tKs8hA6PGpzSXSG8Ca9SAS2mZQaVLbAbbpqrfPDMmdVtgPEw=="

    },

    "IsSuccess": **true**

}

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 404 | 195 | Unable to find card |
| 404 | 147 | Unable to find user |

**Remark:** When the RSA bi-key is generated by the mobile application, the generated key is in PKCS#8 format. And it is in the RSA decryption algo that the padding option in PKCS#1 mode must be specified.

# PIN DISPLAY :

In order to use this feature, you must have a RSA Key couple: RSA public Key and RSA Private Key.

The RSA Keys must be generated in PKCS#8 format ECB and the length must be a least 2048.

## Value uased For the Sandbox :

**Don’t use other values of RSA Keys on SandBox.**

**Don’t use the header and the trailer of the Key, only the body.**

**the Public Key :**

MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArOyRqpUXKcbu3k5sXoYtU3J889qCfOS2pyfGKHlQ/qlBAGilw5kI6YUobNZFfI/iIYZZxIi4OLwLNAzvKJwXHAIYBpczsfB1Y5OOU2QEtCXL3ky00kt9DUSfFhTiiwlzDxZJr6iRby6Sy0nErJ7x9wN/X3Og249djdj3k9VMSdL/efshFriqDDrwLwu1J2YqTpWxMOP7VfYT29ud+gykmM2R5HMke9+lfMfOL2eld1WUPRgCJKwBi8i5ONFI6cUzrkBQHM9EE5Z0PjOaZq7BCXza1wXxbpA0KRtEOe3eZB13sS//LsgAjedAtZMHHTrWsUfHEP+N5YUbzeGW+EkIsQIDAQAB

**the Private Key :**



## Request :

Request : (GET) /api/v1.0/users/{appuserid}/cards/{appcardid}/PINDisplay? ChannelCode=XX&PublicKey=YY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| AppCardId | String | X(255) | Xpollens\_CP3 | Card External Reference |
| ChannelCode | String | An2 | 66 | Channel code |
| PublicKey | String | AN256...800 (base 64) | MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArOyRqpUXKcbu3k5sXoYtU3J889qCfOS2pyfGKHlQ/qlBAGilw5kI6YUobNZFfI/iIYZZxIi4OLwLNAzvKJwXHAIYBpczsfB1Y5OOU2QEtCXL3ky00kt9DUSfFhTiiwlzDxZJr6iRby6Sy0nErJ7x9wN/X3Og249djdj3k9VMSdL/efshFriqDDrwLwu1J2YqTpWxMOP7VfYT29ud+gykmM2R5HMke9+lfMfOL2eld1WUPRgCJKwBi8i5ONFI6cUzrkBQHM9EE5Z0PjOaZq7BCXza1wXxbpA0KRtEOe3eZB13sS//LsgAjedAtZMHHTrWsUfHEP+N5YUbzeGW+EkIsQIDAQAB | Rsa public key 2048 |

**Example:**

https://sb-api.xpollens.com/api/v1.0/users/xpollens1/cards/xpollens\_CP1/PINDisplay?ChannelCode=66&PublicKey=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArOyRqpUXKcbu3k5sXoYtU3J889qCfOS2pyfGKHlQ/qlBAGilw5kI6YUobNZFfI/iIYZZxIi4OLwLNAzvKJwXHAIYBpczsfB1Y5OOU2QEtCXL3ky00kt9DUSfFhTiiwlzDxZJr6iRby6Sy0nErJ7x9wN/X3Og249djdj3k9VMSdL/efshFriqDDrwLwu1J2YqTpWxMOP7VfYT29ud+gykmM2R5HMke9+lfMfOL2eld1WUPRgCJKwBi8i5ONFI6cUzrkBQHM9EE5Z0PjOaZq7BCXza1wXxbpA0KRtEOe3eZB13sS//LsgAjedAtZMHHTrWsUfHEP+N5YUbzeGW+EkIsQIDAQAB

## Response :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| PinBlock | String | *AN…12* | D5DRGj6F1po= | PIN encrypted with KTA Key |
| KtaKey | String | *AN…24* | g7JDPvyJfB1QdGcXD3Inqw== | KTA key encrypted with KTK Key |
| KtkKey | String | *AN…24* | WJcfDICWyhpQAl7Rzg/9sBImpWCHpA7cOHm/SbRLg4P0XF/dZEzGPX58iCg1A6I36vX6j2cA7mAksWHaA0rTHtAlSCpcm1zNeAoYC7ZiBHUOVXJjThPix2YMHUT15WCn3OrcOcqFHTOoDwRu2t+RHJTUOvzN2RiXe6lDWx0D2GW8euCs6JguucrjbwvzITs9Kf9IFG98f3lUE2eUar5QPBP9CzsVM1+Xe1mIsNSK/+QERXDOyl4tgyU035h5Ot75EglExM6ullYUBlHC9VTG5Ls7UMkRf4VZKQYSsRiNelaQt33+q7YzhHWWtIs1HAo5NLdoO0AI/iEoln7ZMOWi3g== | KTK Key encrypted with RSA public key |

**Example:**

HTTP/1.1 200 CREATED

{

    "PinBlock": "D5DRGj6F1po=",

    "KtaKey": "g7JDPvyJfB1QdGcXD3Inqw==",

    "KtkKey": "WJcfDICWyhpQAl7Rzg/9sBImpWCHpA7cOHm/SbRLg4P0XF/dZEzGPX58iCg1A6I36vX6j2cA7mAksWHaA0rTHtAlSCpcm1zNeAoYC7ZiBHUOVXJjThPix2YMHUT15WCn3OrcOcqFHTOoDwRu2t+RHJTUOvzN2RiXe6lDWx0D2GW8euCs6JguucrjbwvzITs9Kf9IFG98f3lUE2eUar5QPBP9CzsVM1+Xe1mIsNSK/+QERXDOyl4tgyU035h5Ot75EglExM6ullYUBlHC9VTG5Ls7UMkRf4VZKQYSsRiNelaQt33+q7YzhHWWtIs1HAo5NLdoO0AI/iEoln7ZMOWi3g=="

}

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 704 | Input parameter missing or not correct |
| 404 | 195 | Unable to find card |

## decryption of block pin (Method proposed)

The PinBLOCK, KTA, KTK data transmitted in the query response are encoded in Base64 :

**Step 1 : Decryption of the KTK[Kpub] :**

**Take the KTK[Kpub] key from the encoded response in Base64:**

WJcfDICWyhpQAl7Rzg/9sBImpWCHpA7cOHm/SbRLg4P0XF/dZEzGPX58iCg1A6I36vX6j2cA7mAksWHaA0rTHtAlSCpcm1zNeAoYC7ZiBHUOVXJjThPix2YMHUT15WCn3OrcOcqFHTOoDwRu2t+RHJTUOvzN2RiXe6lDWx0D2GW8euCs6JguucrjbwvzITs9Kf9IFG98f3lUE2eUar5QPBP9CzsVM1+Xe1mIsNSK/+QERXDOyl4tgyU035h5Ot75EglExM6ullYUBlHC9VTG5Ls7UMkRf4VZKQYSsRiNelaQt33+q7YzhHWWtIs1HAo5NLdoO0AI/iEoln7ZMOWi3g==

**Step2 : convert it to ByteArray:**

**Step3 : Decipher the KTK[Kpub] with the PKCS padding option #1:**

**Step4 :** **Convert the result in step 3 into hexa (through a ByteArray Table):**

**We get then :**

KTK key decrypted by the private RSA key with diversification:

00006110091288919999**97FD79326EE6E30D80F120706D6825D6**

Value of the KTK = **97FD79326EE6E30D80F120706D6825D6**

Diversifying = 611009128891

***Remark :***

***Decipher KTK[Kpub] with its private RSA key, and check the proper flow of decryption by verifying that the banners framing the Diversifiant are well equal to '0000' and '9999'. If NOK then we don't go any further.***

***Recover the Diversifiant and The Value of the KTK Key in 3DES (The KTK's clear value begins at the end of the banner containing the diversifier. The diversifiant has a length of 12 digital characters. Including the '0000' and '9999' frame, the banner has a total length of 20 digital characters). This diversifier is used for deciphering PinBlock in the XOR application.***

**Step5 : Decryption of the KTA[KTK]**

**3DES to apply :**

**Decryption mode: ECB**

**Padding : None**

**We have the response in the API :KTA [KTK] :**

g7JDPvyJfB1QdGcXD3Inqw==

**For this step, do not convert the KTA [KTK] into hexa for the 3DES application of the step 6. Use a byte array.**

**Step6 : Apply a 3DES decryption to get the KTA deciphered (in hexa):**

6B57FDAECBABB3A732F7A437EF435D89

**Step7 : Decryption of the PinBlock**

From the output in base 64 in the response: D5DRGj6F1po

The value of PinBlock in hexa: 0F90D11A3E85D69A

***Etape 1 :*** Deciphering by applying the 3DES algorithm with clear and encoded KTA in ISO0 format: 0455E5EFF6ED776E

***RemarK :***

***The first character (highlighted in yellow) of pinBLOCK is necessarily '0', if NOK is that the decryption went wrong and in this case we do not go any further.***

***The second character (highlighted in green) of PINBLOCK is the length of PIN***

***Etape 2 :*** Get PINBLOCK in clear (Step 2)

* The application makes one or exclusive (XOR) between the PINBLOCK decrypted in stage 1 and the concatenation of "0000" and the diversified recovered from the decryption stage **KTK[Kpub]** (611009128891)
* A field containing "0" is then obtained, followed by the length of the PIN, followed by the PIN and supplemented by FF characters to have 16 characters
* *If the PADding to FF is not present (or does not start just after the length of the PIN: for a PIN length equal to 4 in our case, the padding must start from the 6th character), it's NOK we do not go any further*
* **PINBLOCK decrypted in 3DES with KTA (KTAENCLAIR): 0455E5EFF6ED776E (encoded with standard ISO0)**
* **After XOR: 045584FFFFFFFF**
* **Card’s PIN: 5584**

# Card SelfCare

The card selfcare allows to:

* Modify the ATM weekly limit
* Modify the card Payment monthly limit
* Block a card
* Block internet payment
* Bock Foreign payment

## Selfcare V1.0/ V1.1

**Remark**: There’ s no swagger available for the car order V1.0/ V1.1

### Request V1.0/ V1.1

To update one of the attributes above, we should request the following resource using an http put request:

**Put**: / api/v1.0/users/{appuserid}/cards/{appcardid}/

**Body:**

You should encapsulate the following JSON in the body of the request to update one or the whole attributes.

{

"CardLimits": {

        "ATMWeeklyAllowance": 200,    (optional)

        "MonthlyAllowance": 800,     (optional)

    },

    "Blocked": 0, (optional)

    "ForeignPaymentBlocked": 1, (optional)

    "InternetPaymentBlocked": 1 (optional)

}

### Response

When the request is correct a 200 http response will be received, and the card selfcare details are returned.

Otherwise, an error is raised.

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 404 | 195 | Unable to find card |
| 404 | 147 | Unable to find user |
| 404 | 1066 | Incorrect limit value |

Example:

{

    "Name": "Physical card ",

    "CardLimits": {

        "Transaction": **null**,

        "ATMWeeklyAllowance": 200,

        "ATMWeeklyUsedAllowance": 0,

        "MonthlyAllowance": 800,

        "MonthlyUsedAllowance": 0

    },

    "Blocked": 0,

    "ForeignPaymentBlocked": 1,

    "InternetPaymentBlocked": 1

}

## Selfcare V2.0

swagger available for the car order V2.0 on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

For the globalLimitAtmSelected and globalLimitPaymentSelected, the an be done in the limit of the offer and for the lower limit must be an interger >0.

### Request V2.0

To update one of the attributes above, we should request the following resource using an http put request:

Request: PUT/ /api/v2.0/card/{cardExternalRef}

{

  "globalLimitAtmSelected": 250, (optional)

  "globalLimitPaymentSelected": 900, (optional)

  "cardBlocked": **true**, (optional)

  "isVadBlocked": **true**, (optional)

  "foreignPaymentBlocked": **true** (optional)

}

### Response V2.0

Response v2.0 : 200 OK

{

    "label": "Physical card",

    "isAlphaTest": **false**,

    "cardExternalRef": "XPOLLENS1\_XX",

    "status": {

        "cardStatusCode": "SENT",

        "description": "Sent"

    },

    "oppositionReasonCode": "",

    "creationDate": "2020-08-12T09:14:41.0770573",

    "expiryDate": "09-2023",

    "visualCodeSelected": "SEZR",

    "isBlocked": **true**,

    "globalLimitAtmSelected": 250,

    "globalLimitPaymentSelected": 900,

    "holder": {

        "holderExternalRef": "xpollens1",

        "partnerCode": "Partne1",

        "holderFirstName": "jordan",

        "holderLastName": "martin",

        "holderEmail": "xpollens123@s-money.fr",

        "holderPhoneNumber": "+33645799655"

    },

    "offerPartner": {

        "offerPartnerCode": "ExampleClassicPhysical",

        "partnerCode": "Partner1"

    },

    "uniqueId": "1652811111+NJL0UirckOqr8M8u9S/Yg",

    "bankId": 30007,

    "hint": "1111XXXXXXXX6238",

    "isGeoblocked": **true**,

    "isVadBlocked": **true**,

    "wishPin": **false**,

    "oldExternalRef": **null**,

    "cancellationReasonCode": "OtherReason"

}

# Card CancelLation V2.0

Swagger available on the sandbox environment: [https://sb-api.xpollens.com/swagger/index.html](https://sb-api.xpollens.com/swagger/index.html" \t "_blank" \o "https://sb-api.xpollens.com/swagger/index.html)

This feature is only available in V2.0.

### Request Cancels a card v2.0

Request: (PATCH) /api/v2.0/card/{cardExternalRef}/cancel?cancellationReason=1

List of different values of the ‘CancellationReason’:

**1 : the customer no longer wants his card**  
**2 : the customer left the bank**  
**3 : deceased customer**  
**4 : Partner request**  
**6 : other reason (default value if not provided in the input parameters)**

### Response Cancels a card v2.0

Response v2.0 : 204 OK

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 006 | The card does not exist |

# Card Opposition V2.0

Swagger available on the sandbox environment: https://sb-api.xpollens.com/swagger/index.html?urls.primaryName=CardFactory%20API%20-%20v2.0#/Card/patch\_api\_v2\_0\_card\_\_cardExternalRef\_\_oppose

This feature is only available in V2.0.

### Request Opposes a card v2.0

PATCH/api/v2.0/card/{cardExternalRef}/oppose

{

  "oppositionReasonCode": 34, *(mandatory)*

  " channelCode ": 4, *(mandatory)*

  "oppositionDeclarationDate": "2020-10-16T07:54:35.948Z" *(Optional)*

}

### Response Opposes a card v2.0

Response v2.0 : 200 OK

### Error codes

|  |  |  |
| --- | --- | --- |
| **http code** | **Error code** | **Description** |
| 400 | 001 | Mandatory parameter are missing |
| 400 | 017 | Existing opposition |

# CVX2 DISPLAY V2.0

Request for the card’s Cvx2 fOr only virtual cards using a new API V2.0.

The values in the answer are encrypted in RSA using the public key given in the request.

Request:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| ChannelCode | String | An2 | 66 | Channel code;  List of values:  04 = by computer  66 = by phone  72 = by tablet |
| publicKey | String | AN256...800 (base 64) | MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArOyRqpUXKcbu3k5sXoYtU3J889qCfOS2pyfGKHlQ/qlBAGilw5kI6YUobNZFfI/iIYZZxIi4OLwLNAzvKJwXHAIYBpczsfB1Y5OOU2QEtCXL3ky00kt9DUSfFhTiiwlzDxZJr6iRby6Sy0nErJ7x9wN/X3Og249djdj3k9VMSdL/efshFriqDDrwLwu1J2YqTpWxMOP7VfYT29ud+gykmM2R5HMke9+lfMfOL2eld1WUPRgCJKwBi8i5ONFI6cUzrkBQHM9EE5Z0PjOaZq7BCXza1wXxbpA0KRtEOe3eZB13sS//LsgAjedAtZMHHTrWsUfHEP+N5YUbzeGW+EkIsQIDAQAB | Rsa public key 2048 |

Sample request:

Request : (POST) /api/v2.0/cvx2/{appcardid}

{

"channelCode": "04",

"publicKey": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxzvuvn10db/hAY5BIxe1YFNDYeuAz0LqhKA8tEK/1rTR4dnvHVwYHkY0ZCEhJBYlJGBJsjasBSqacqHhJjMsEGWURp84raXIU0O/LRcGcQIpBmoe/AKacPyWGpCBM5Btw6hh/QinptXUNWtjhRjzJykSmwyBp8duH9nJDXWvbADYlZ69IFXvIoC16DKYVqJvqu7wNolcMHUwkgrRu7xJBGs+P+nbFrlkhR4wiudcZwgYHG4foTZnHY3or2JJbX55Gr1CxXgLZ2rpaLdprgZBbTuBNj3KDaEiP+QEACC+YsUhbmbt7YpygQ2qKQQPPHjh3hdcTLz1WnKHz5IB8kzmOQIDAQAB"

}

Response:

The response is composed of an http status code, and data in the body.

In normal flow without errors, these are the expected response fields:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| cvx2 | string | X(3) | {      "cvx2": "Vag8396ok2faNe40GA+3wXRvC9OPH9zZ8sJzCRjGG2KqIR4ibcymqSy9FVl6BjZvEl3JkH/utBX1oRuPywRzK4rhu8EMh8ErBopsamFTLEnxaumz9gnT+oOp2Bqt7aYTNwOMOrkemT4vMj/eC9yMJ1yqNSQgYy3tOIsnptDTxLXYDyAmgSofzFLf0cEh1+4MJdWpdQI4rK7u/gevlbn6uGL5/tcobntZO9ZESyTMgNzn7aLkXh7ZzKpYU9QI2EaHSlsrmKv6dgwLeUBqv8cTudbX9v4/T0qvkvDhSs1JdX1sMvTSZE1qjq7v07NI5E3EqX3jaGMm7DOKFNX67QXJ4A=="  } |  |

* <Buffer\_to\_encrypt> : json message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| alea | string | X(16) | qD1ey9Z9p0X2X2cV | Alea (can be ignored) |
| Algo | string | X(4) | VISA |  |
| cvx2 | string | X(3) | 506 | Card Verification Code |

Sample WS response (the body):

HTTP/1.1 201 OK

{

    "cvx2": "Vag8396ok2faNe40GA+3wXRvC9OPH9zZ8sJzCRjGG2KqIR4ibcymqSy9FVl6BjZvEl3JkH/utBX1oRuPywRzK4rhu8EMh8ErBopsamFTLEnxaumz9gnT+oOp2Bqt7aYTNwOMOrkemT4vMj/eC9yMJ1yqNSQgYy3tOIsnptDTxLXYDyAmgSofzFLf0cEh1+4MJdWpdQI4rK7u/gevlbn6uGL5/tcobntZO9ZESyTMgNzn7aLkXh7ZzKpYU9QI2EaHSlsrmKv6dgwLeUBqv8cTudbX9v4/T0qvkvDhSs1JdX1sMvTSZE1qjq7v07NI5E3EqX3jaGMm7DOKFNX67QXJ4A=="

}

Encrypted buffer content (json):

{"alea":"YCmdLK6lzzqvSYrG","algo":"VISA","cvx2":"922"}

# PAN DISPLAY V2.0

Request for the card’s PAN for only activated cards (Virtual or Physical) using a new API V2.0.

The values in the answer are encrypted in RSA using the public key given in the request.

Request:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Exemple** | **Description** |
| ChannelCode | String | An2 | 66 | Channel code;  List of values:  04 = by computer  66 = by phone  72 = by tablet |
| publicKey | String | AN256...800 (base 64) | MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArOyRqpUXKcbu3k5sXoYtU3J889qCfOS2pyfGKHlQ/qlBAGilw5kI6YUobNZFfI/iIYZZxIi4OLwLNAzvKJwXHAIYBpczsfB1Y5OOU2QEtCXL3ky00kt9DUSfFhTiiwlzDxZJr6iRby6Sy0nErJ7x9wN/X3Og249djdj3k9VMSdL/efshFriqDDrwLwu1J2YqTpWxMOP7VfYT29ud+gykmM2R5HMke9+lfMfOL2eld1WUPRgCJKwBi8i5ONFI6cUzrkBQHM9EE5Z0PjOaZq7BCXza1wXxbpA0KRtEOe3eZB13sS//LsgAjedAtZMHHTrWsUfHEP+N5YUbzeGW+EkIsQIDAQAB | Rsa public key 2048 |

Sample request:

Request : (POST) /api/v2.0/pan/{appcardid}

{

"channelCode": "66",

"publicKey": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxzvuvn10db/hAY5BIxe1YFNDYeuAz0LqhKA8tEK/1rTR4dnvHVwYHkY0ZCEhJBYlJGBJsjasBSqacqHhJjMsEGWURp84raXIU0O/LRcGcQIpBmoe/AKacPyWGpCBM5Btw6hh/QinptXUNWtjhRjzJykSmwyBp8duH9nJDXWvbADYlZ69IFXvIoC16DKYVqJvqu7wNolcMHUwkgrRu7xJBGs+P+nbFrlkhR4wiudcZwgYHG4foTZnHY3or2JJbX55Gr1CxXgLZ2rpaLdprgZBbTuBNj3KDaEiP+QEACC+YsUhbmbt7YpygQ2qKQQPPHjh3hdcTLz1WnKHz5IB8kzmOQIDAQAB"

}

Response:

The response is composed of an http status code, and data in the body.

In normal flow without errors, these are the expected response fields:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property | Type | Format | Example | Description |
| cvx2 | string | X(3) | {      "encryptedData": "Mja8EoxelOZb/XTaxEkCFyT1TDb//7PoGoL+uq+VvxVckII9y7b3oiiVkoaFKtgTxCPW3/l2FJ+06fveFoW3oMFnnSv7NA8gDpS31VJfSoGZQmn2gg/DlEMrnImCKUi/aRu6oP5cHIzSzeO08iqUAnKKL24rvdzuIeD2xZy0ouMUmqaRL5JQd42R0COIiOkkmXyd/VXjQNsJK9mJdqf4EZ+QnjVD8v5BBHiD+8nGBtlU4Q9NJJ0+lm2LUVlyMODZoxv9whYuGFh5X2ESuarU3K966pSBExspac1Uc5aoBOyC25phk+FLNpMnBgtXbQvicYpEGTQuejbc83uA5PVScQ=="  } |  |

* <Buffer\_to\_encrypt> : json message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Property** | **Type** | **Format** | **Example** | **Description** |
| primaryAccountNumber | string | X(16) | 1234042345678901 | PAN |
| cardExpirationDate | string | X(4) | 1023 | Card Expiration date |

**Sample WS response (the body):**

Sample WS response (the body):

HTTP/1.1 201 OK

{

    "encryptedData": "Mja8EoxelOZb/XTaxEkCFyT1TDb//7PoGoL+uq+VvxVckII9y7b3oiiVkoaFKtgTxCPW3/l2FJ+06fveFoW3oMFnnSv7NA8gDpS31VJfSoGZQmn2gg/DlEMrnImCKUi/aRu6oP5cHIzSzeO08iqUAnKKL24rvdzuIeD2xZy0ouMUmqaRL5JQd42R0COIiOkkmXyd/VXjQNsJK9mJdqf4EZ+QnjVD8v5BBHiD+8nGBtlU4Q9NJJ0+lm2LUVlyMODZoxv9whYuGFh5X2ESuarU3K966pSBExspac1Uc5aoBOyC25phk+FLNpMnBgtXbQvicYpEGTQuejbc83uA5PVScQ=="

}

Encrypted buffer content (json):

{"primaryAccountNumber":"4396100020360907","cardExpirationDate":"0624"}