



## API Fields and File Layouts

Version 3.07

## Version History

Date	Author	Version	Change Description
02/26/2021	Juan C Tellez	1.0	Original version
08/09/2021	Juan C Tellez	2.0	Additional tables
08/20/2021	Juan C Tellez	2.1	Adding file layout
12/14/2021	Juan C Tellez	2.2	Adding fields conditions and API response changes
03/21/20022	Jan C Tellez	2.4	New production endpoint added
08/03/2022	Juan C Tellez	2.5	Compliance Fields
08/27/2022	Juan C Tellez	2.6	Compliance and Risk Fields Update
12/27/2022	Juan C Tellez	2.7	New Production endpoint.
08/27/2022	Juan C Tellez	2.8	Compliance and Risk Fields Update; Remove "Business Activity" field
02/16/2023	Juan C Tellez	2.9	Compliance and Risk Fields Update; Added example
02/21/2023	Juan C Tellez	2.10	Token credentials clarification
03/19/2023	Juan C Tellez	2.11	Include UpdateCardholderCredentials method
03/19/2023	Juan C Tellez	2.11	Include the CardHolderAddCard method.
04/02/2023	Juan C Tellez	2.12	Add support to use PCI compliant environment
04/13/2023	Felipe Tellez	2.13	Add Core Banking System account endpoints
04/18/2023	Juan C Tellez	2.14	Add account to account endpoint
05/06/2023	Juan C Tellez	2.15	Remove not required endpoints
08/01/2023	Juan C Tellez	2.16	Add accounts fields
09/07/2023	Saif Ud Din	2.17	Add iframe Api - Sandbox
11/20/2023	Saif Ud Din	2.18	Add iframe Api - Production
12/12/2023	Felipe Tellez	2.19	Change to PROD endpoints, add Swagger details
12/22/2023	Felipe Tellez	2.20	Change to Sandbox endpoint
02/20/2023	Saif Ud Din	2.21	Add load network endpoint detail
03/27/2023	Saif Ud Din	2.22	Reload via Token and Reload via Barcode
07/23/2024	Felipe Tellez	3.02	Change Cardreferenceid from 110XXXX to 127XXX, update insertcardholder/update profile samples and addition of reissue card sample
09/10/2024	Saif Ud Din	3.03	Add iframe Api – IMT endpoint
10/11/2024	Felipe Tellez	3.04	Add IMT Get Quote endpoint, Add MAP_CardsbyReferece endpoint
10/24/2024	Felipe Tellez	3.05	Update IMT and card load endpoints
12/23/2024	Felipe Tellez	3.06	Update ID document table
01/07/2025	Felipe Tellez	3.07	Addition of OTP endpoints

# Table of Contents

- 1. Staging Endpoint and Test Credentials ..... 5
- 2. Basic operations..... 6
- 3. PRODUCTION Endpoint..... 6
- 4. Card Status Codes ..... 6
- 5. Foreign ID Types ..... 7
- 6. Address Types..... 8
- 7. Gender Codes ..... 8
- 8. File Type..... 8
- 9. Country Codes ..... 9
- 10. Language IDs (LangId) ..... 9
- 11. Reissue Types..... 13
- 12. History/Statement Types ..... 13
- 13. Is Forced Post..... 13
- 14. Apply Fee ..... 14
- 15. Cardholder Fields Conditions..... 14
- 16. Insert Cardholder API – (Sample)..... 16
- 17. InsertCardholder Response (The card was successfully created.) ..... 18
- 18. InsertCardholder Response (declined by KYC)..... 18
- 19. Update Profile (Sample)..... 19
- 20. Update Profile API – *add files* ..... 21
- 21. Reissue Card (Sample) ..... 21
- 22. Compliance and Risk Field Values ..... 22
- 23. UpdateCardholderCredentials - ..... 24
  - 23.1 Endpoint ..... 25
  - 23.1 Request ..... 25
  - 23.1 Response:..... 25
- 24. CardHolderAddCard-..... 26
  - 24.1 Endpoint ..... 26
  - 24.1 Request ..... 26
  - 24.1 Response:..... 26
- 25. Sensitive Card information support ..... 27
  - 25.1 Display Card information. .... 27
  - 25.1 Capture new card pin ..... 28
- 26. Bank Account Endpoints-..... 29
  - 26.1. Create customer’s Account ..... 29
  - 26.2. Map account to cardholder (customer) ..... 29
  - 26.3. Get Core Banking CustomerID by CardholderID ..... 29
  - 26.4. Get Core Banking Accounts by Core Banking CustomerID ..... 30
  - 26.5. Get Core Banking Account’s transactions. .... 31
  - 26.6. Process a Core Banking Account to Card transfer ..... 34
  - 26.7. Process a Card to Core Banking Account transfer ..... 34
  - 26.8. Get Core banking system account beneficiaries. .... 36
  - 26.9. Process a Core Banking Account to Core Banking Account transfer. .... 38
  - 26.10. Account to Account process. .... 39
  - 26.11. New Prospect ..... 41
- 27. Display Card Information (iframe) - Sandbox..... 43
- 28. Capture new pin card (iframe) - Sandbox ..... 44
- 29. Display Card Information (iFrame) - Production ..... 45

---

30. Capture new pin card (iFrame) - Production .....	45
31. Capture Credit Card Information to process a payment - Production .....	46
32. Capture IMT Information to process a Send Money transaction - Sandbox .....	48
33. Reload Card – ATM Cash In .....	50
34. Reload Card – ATM Locator .....	50
35. Reload Card – Barcode Cash In .....	51
36. IMT Get Quote .....	51
37. IMT Transaction .....	54
38. IMT Get Cities .....	54
39. IMT Get Offices .....	55
40. IMT Cancel Transaction .....	56
41. Program Owner (PO) Webhook .....	56
42. Map Cards by Card ReferenceID .....	57
43. Request One Time Pin (OTP) .....	58
44. Validate One Time Pin (OTP) .....	59
45. Error Codes .....	61

# 1. Staging Endpoint and Test Credentials

Endpoint: [https://apisandbox.nibank.com/BusRest\\_V202401/api/](https://apisandbox.nibank.com/BusRest_V202401/api/)

Credentials:

User: SanboxTest

Password: SNDv1iJ7t?s!WwbZqy+zkrdBOX

Example:

To verify credentials and get the token:

[https://apisandbox.nibank.com/BusRest\\_V202401/api/Admin/VerifyUserCredentials](https://apisandbox.nibank.com/BusRest_V202401/api/Admin/VerifyUserCredentials)

For test purposes you can use

CustomerID = 1

Program ID = 1

*For production we will provide the CustomerID and Program ID.*

Request:

```
{  
  "appType": 1,  
  "user": "SanboxTest",  
  "pass": "SNDv1iJ7t?s!WwbZqy+zkrdBOX",  
  "CustomerID": 1  
}
```

This request returns an authentication token (AccessToken) as part of the response.

```
"AccessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bmRldWVfbmFtZSI6IiNhbWJveFRlc3QiLCJodHRwOi8vc2NoZW1hcy5taWVudC9yY29tL3dzLzlwMDgvMDYvaWRlbnRpdHkvY2xhaW1zL3VzZXJkYXRhIjoiMzciLCJkb2N0b21lcklEiwoiMSIsIm5iZiI6ImY3NzA2MjMyMSwiZXhwIjoxNjc3MTA1NTIwLCJpYXQiOiJlNzIwMjcwNjIzIj0iLCJ0eXAiOiJKV1QiLCJkaXI6Imh0dHA6Ly9sb2NhbGhvc3QvliwiYXVkljoiaHR0cDovL2xvY2FsaG9zdC8ifQ.-Gx5qxmZm4X7XAhpGRvx4FdGFsjeit-FVykq55gmYms"
```

The AccessToken must be passed as a header to the API's that require it together with a string Bearer, the result would be the following:

Request Headers

Content-Type: application/json



Code	Abbreviation	Description
A	ISSUED INACTIVE	Card has been issued but not yet activated
B	OPEN	All transactions allowed
C	LOST CARD	Lost card
D	STOLEN CARD	Stolen card
E	NO WITHDRAWALS	No withdrawals are allowed to be made on card. (Restricted)
F	CLOSED	Card account has been closed
G	LOST NOT CAP	Lost card not captured
H	STOLEN NOT CAP	Stolen card not captured
I	INACTIVE	Card is inactive or blocked
R	CARD REISSUE	Card reissue
S	Fraud Block	Card blocked due to fraud

## 5. Foreign ID Types

The following table provides the different types of Foreign IDs along with their descriptions.

Foreign ID Type	Foreign ID Description
1	Driver's License
2	ID Card/Guatemala DPI
3	Passport
4	Consular ID/Mexican Matrícula Consular
5	Other ID
6	SSN
7	Alien Registration Card
8	El Salvador ID Card (DUI)
9	Nicaragua ID Card
10	Federal ID Number

12	US Military ID Card
11	CNPJ Tax Payer Identification Number (Only for Brazil)
13	CPF <i>Cadastro de Pessoas Físicas</i> (Natural Persons Register Only for Brazil)

## 6. Address Types

The following table provides the different types of Address's along with their descriptions.

Address Type	Address Type Description
1	Shipping Address
2	Billing Address

## 7. Gender Codes

The following table provides the different types of Gender Code along with their descriptions.

Gender Code	Gender Code Description
0	Male
1	Female
3	Other
null	Other

## 8. File Type

The following table provides the different types of File types along with their descriptions.

File Type Code	File type Description
1	National ID
2	Driver's License
3	Utility Bill
4	Passport



5	Other
---	-------

## 9. Country Codes

The country codes must be aligned with ISO 3166 (alpha-2 or alpha-3 codes depending on the field length.)

## 10. Language IDs (LangId)

This table lists the language options, and their corresponding identification codes, for the cardholder application that may be supported by the client and opted by the cardholder.

Srno	LangId	Description
1	en	English
2	es	Español
3	pt	Português
4	en_US	English/US
5	es_US	Español/US
6	ja	japanese
7	de	German
8	en_ZA	English South Africa
9	fr	French
10	zh	Chinese
11	it	Italian
12	pl	Polish
13	sv	Swedish
14	cs	Czech
15	af	kaans
16	sq	Albanian
17	am	Amharic

Srno	LangId	Description
18	ar	Arabic - Algeria
19	hy	Armenian
20	as	Assamese
21	az	Azeri - Cyrillic
22	eu	Basque
23	be	Belarusian
24	bn	Bengali - Bangladesh
25	bs	Bosnian
26	bg	Bulgarian
27	my	Burmese
28	ca	Catalan
29	hr	Croatian
30	da	Danish
31	Dhivehi	Divehi
32	nl	Dutch - Belgium
33	et	Estonian
34	mk	FYRO Macedonia
35	fo	Faroese
36	fa	Farsi - Persian
37	fi	Finnish
38	gd	Gaelic - Ireland
39	gl	Galician
40	ka	Georgian
41	el	Greek

Srno	LangId	Description
42	gn	Guarani - Paraguay
43	he	Hebrew
44	hi	Hindi
45	hu	Hungarian
46	is	Icelandic
47	id	Indonesian
48	kn	Kannada
49	ks	Kashmiri
50	kk	Kazakh
51	km	Khmer
52	ko	Korean
53	lo	Lao
54	la	Latin
55	lv	Latvian
56	lt	Lithuanian
57	ml	Malayalam
58	mt	Maltese
59	mi	Maori
60	mr	Marathi
61	mn	Mongolian
62	ne	Nepali
63	nb	Norwegian - Bokml
64	nn	Norwegian - Nynorsk
65	or	Oriya

Srno	LangId	Description
66	rm	Raeto-Romance
67	ro	Romanian - Moldova
68	ru	Russian
69	sa	Sanskrit
70	sr	Serbian - Cyrillic
71	tn	Setsuana
72	sk	Slovak
73	sl	Slovenian
74	so	Somali
75	sb	Sorbian
76	sw	Swahili
77	tg	Tajik
78	ta	Tamil
79	tt	Tatar
80	te	Telugu
81	th	Thai
82	bo	Tibetan
83	tr	Turkish
84	tk	Turkmen
85	uk	Ukrainian
86	ur	Urdu
87	uz	Uzbek - Cyrillic
88	vi	Vietnamese
89	cy	Welsh

Srno	LangId	Description
90	xh	Xhosa
91	yi	Yiddish

## 11. Reissue Types

Re issue card type.

Reissue type	type Description
1	Same Card Number with Same profile Info.
2	Same Card Number with Updated Expiry.
3	New Card Number with existing profile info.

## 12. History/Statement Types

The following table provides the possible values for the History/Statement Types field.

Type	Description
A	Include only Pending Preauthorization transactions in Transaction History & MiniStatement services
P	Include only Posted transactions in Transaction History & MiniStatement services
B	Include both Pending Preauthorizations & Posted Transactions in Transaction History & MiniStatement services

## 13. Is Forced Post

The IS Forced Post field is used to force a transaction without considering the status of the card.

Important: The default value for this field should be false. A true value should **only be used in exceptional circumstances**, using the true value could override any card status or card balance.

Type	Description
False	The transaction will be processed only if the card status allows the transaction to be processed and/or if the
True	The transaction will override any card restriction. DO NOT USE this value except if you are sure you want to override the card status

## 14. Apply Fee

Indicates whether the transaction fee should be deducted.

Type	Description
False	The fee will NOT be deducted
True	The fee will be deducted

## 15. Cardholder Fields Conditions

The following table provides the Cardholder fields conditions.

Field	Len
City	STR 60
Country	AN 3
County	STR 30
Line1	STR 60
Line2	STR 60
Line3	STR 60
State	STR 60
ZipCode	AN 80
CellNumber	N 15
CellPhoneProvider	STR 60
DateOfBirth	D 10
DefaultLanguage	A 2
DrivingLicenseNumber	AN 21
DrivingLicenseState	A 2
Email	STR 60
EmployerAddress	STR 60
EmployerName	STR 60
EmployerTelephone	STR 60
EmployerYears	STR 60
FirstName	STR 60
LastName	STR 60
MiddleName	STR 60
MotherMaidenName	STR 60
ForeignCountryCode	AN 3

ForeignID	AN 30
ForeignIDType	AN 2
Gender	A 1 Can be either 'M' or 'F' Where M = Male, F = Female
IDExpDate	D 10
IDType	AN 2
MartialStatus	AN 30
NationalID	AN 30
NationalIDIssueCountry	AN 3
Nationality	AN 2
Occupation (please refer to section "Compliance and Risk Field Values")	AN 30
PlaceOfBirth	AN 30
ResidentialPhone	N 15
Surname	AN 3
WorkPhone	N 15
Customer Subtype (please refer to section "Compliance and Risk Field Values")	AN 10
Source of Funds (please refer to section "Compliance and Risk Field Values")	AN 10
Source of Wealth (please refer to section "Compliance and Risk Field Values")	AN 10
Account_Purpose (please refer to section "Compliance and Risk Field Values")	AN 10
Account_Type (please refer to section "Compliance and Risk Field Values")	AN 10
Product_Type (please refer to section "Compliance and Risk Field Values")	AN 10
ET_Amount_Incoming (please refer to section "Compliance and Risk Field Values")	AN 10
ET_Amount_Outgoing (please refer to section "Compliance and Risk Field Values")	AN 10
ET_Quantity_Incoming (please refer to section "Compliance and Risk Field Values")	AN 10
ET_Quantity_Outgoing (please refer to section "Compliance and Risk Field Values")	AN 10
FI_Account_Number (please refer to section "Compliance and Risk Field Values")	AN 10
FI_Financial_Institution (please refer to section "Compliance and Risk Field Values")	AN 10
FI_Currency (please refer to section "Compliance and Risk Field Values")	AN 10
Fatca_US (please refer to section "Compliance and Risk Field Values")	AN 10
Fatca_Tax_ID (please refer to section "Compliance and Risk Field Values")	AN 10

Fatca_Third_parties (please refer to section "Compliance and Risk Field Values")	AN 10
--	-------

## 16. Insert Cardholder API – (Sample)

Please see below an example of the insertcardholder method used to create a new card

```
{
  "programId": 1,
  "cardholder": {
    "AddedBy": 1,
    "BillingAddress": {
      "AddressType": 1,
      "CardHolderAddressId": 1,
      "City": "Miami",
      "Country": "USA",
      "County": "Dade",
      "Line1": "401 Main Street",
      "Line2": "Apt 101",
      "Line3": null,
      "State": "FL",
      "Status": 1,
      "ZipCode": "33178"
    },
    "CardLogId": 2,
    "CardNumber": null,
    "CardProgramId": 4,
    "CardUsage": 1,
    "CardholderId": 0,
    "CustomerId": 1,
    "IPAddress": "10.168.1.1",
    "NameOnCard": null,
    "PersonalAddress": {
      "AddressType": 1,
      "CardHolderAddressId": 1,
      "City": "Miami",
      "Country": "USA",
      "County": "Dade",
      "Line1": "401 Main Street",
      "Line2": "Apt 101",
      "Line3": null,
      "State": "FL",

```



```
"Status": 1,
"ZipCode": "33178"
},
"PersonalData": {
  "AlternateID": null,
  "CardType": null,
  "CellNumber": "3102089819",
  "CellPhoneProvider": null,
  "DateOfBirth": "2000-07-11",
  "DefaultLanguage": "1",
  "DrivingLicenseNumber": null,
  "DrivingLicenseState": null,
  "Email": "robert_test@test.com",
  "EmployerAddress": null,
  "EmployerName": "B",
  "EmployerTelephone": "1133",
  "EmployerYears": null,
  "FirstName": "Robert",
  "ForeignCountryCode": "COL",
  "ForeignID": "P345664322",
  "ForeignIDType": "3",
  "Gender": 1,
  "IDExpDate": "2024-07-11T13:03:14.8992039+00:00",
  "IDType": null,
  "LastAuthentication": "2024-07-11T13:03:14.8992039+00:00",
  "LastName": "Smith",
  "LastSanctionsReview": "2024-07-11T13:03:14.8992039+00:00",
  "MaritalStatus": "Single",
  "MiddleName": "M",
  "MotherMaidenName": null,
  "NationalID": null,
  "NationalIDIssueCountry": null,
  "Nationality": "CO",
  "Occupation": "41",
  "Password": null,
  "PasswordStatus": null,
  "PlaceOfBirth": null,
  "ResidentialPhone": "3102089812",
  "Signature": null,
  "Status": 1,
  "Surname": "Mr",
  "UniqueKey": null,
  "User": null,
  "WorkPhone": "3102089812"
},
"Referenceld": null,
"IstFiles": [
  {
```

```

      "CardholderID": 1,
      "FileContent":
"iVBORwOKGgoAAAANSUUhEUgAAAj8AAAD2CAYAAAA51OziAAAAGXRFWHRTb2Z0d2FyZQBBZG9iZCYII=",
      "FileID": 2,
      "FileName": "99_20181116142659_NIB.png",
      "FileType": 4
    },
    {
      "CardholderID": 1,
      "FileContent": "/9j/4AAQSkZJRgABAQAAQABAADBH//Z",
      "FileID": 2,
      "FileName": "images.jpg",
      "FileType": 4
    }
  ]
}
}

```

You can include as many documents as you need. Minimum 2 are mandatory, the ID image and the proof of address image. The max size for each document is 2 Megabytes.

## 17. InsertCardholder Response (The card was successfully created.)

Please see below the “InsertCardHolder” API response when the card was successfully created.

```

{
  "BatchReferenceID": "8312 - 127007854842",
  "CardholderId": 4020,
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Uploaded Files"
  },
  "NameOnCard": "Robert Smith",
  "ProgramName": "__MG USA",
  "TransId": "F15356569",
  "TransactionCompleted": true
}

```

## 18. InsertCardholder Response (declined by KYC)

The “InsertCardHolder” API response when it is declined because of any compliance reason.

```

{
  "BatchReferenceID": "NO BATCH - NO CARD CREATED",
  "CardholderId": 0,
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "9001",

```

```

    "ReturnDesc": "Success. - application rejected per KYC evaluation - Log ID: 6784 -- Uploaded Files OK"
  },
  "NameOnCard": "null",
  "ProgramName": "PGM-1-TEST",
  "TransId": "null",
  "TransactionCompleted": false
}

```

**Notes:**

- The submitted files will be saved on the system.

## 19. Update Profile (Sample)

Please see below an example of the updateprofile method. This method is used to update the cardholder information for when cards have been previously generated in bulk.

```

{
  "programId": 1,
  "UpdateProfile1": {
    "ApplyFee_Setting": true,
    "CardReferenceId": "127007854842",
    "CardholderID": "4020",
    "CustomerID": 1,
    "Profile": {
      "Address": "401 Main Street",
      "Address2_Optional": "Apt 101",
      "BillingCity_Optional": "Miami",
      "BillingAddress1_Optional": "401 Main Street",
      "BillingAddress2_Optional": "Apt 101",
      "BillingAddress3_Optional": null,
      "BillingAddress4_Optional": null,
      "BillingAddress5_Optional": null,
      "BillingCountry_Optional": "USA",
      "BillingState_Optional": "FL",
      "BillingZipOrPostalCode_Optional": "33178",
      "CardType": null,
      "CellNumber": "3102089819",
      "CellPhoneProvider": null,
      "City": "Miami",
      "Country": "USA",
      "DOB": "2000-07-11",
      "DefaultLanguage": "1",
      "DrivingLicenseNumber_Optional": null,
      "DrivingLicenseState_Optional": null,
      "Email": "robert_test@test.com",
      "EmployerAddress": null,
      "EmployerName": "6",
      "EmployerTelephone": "1133"
    }
  }
}

```

```
"EmployerYears": null,
"FirstName": "Robert",
"ForeignCountryCode_Optional": "COL",
"ForeignIDType_Optional": "3",
"ForeignId_Optional": "P345664322",
"Gender_Optional": "1",
"IDExpDate": "2024-07-11T13:03:14.8992039+00:00",
"IDType": null,
"LastName": "Smith",
"MaritalStatus": "Single",
"MiddleName_Optional": "M",
"MotherMaidenName_Optional": null,
"NameOnCard": null,
"Notes": null,
"Occupation": "41",
"PostalCode": "33178",
"ResidentialPhone": "3102089812",
"SSN_Optional": null,
"ShipMethod": null,
"ShipToFirstName": null,
"ShipToLastName": null,
"Signature": null,
"StateCode": null,
"Surname": "Mr",
"SurnameOther": null,
"WorkPhone_Optional": "3102089812"
},
"UpdateAllCustomerCards_Setting": true,
"IstFiles": [
  {
    "CardholderID": 1,
    "FileContent": "iVBORw0KGgoAAAANSUheUgAAAJ8AAAD2CAYAAAA51OZiAAAAGXRYII=",
    "FileID": 2,
    "FileName": "99_20181116142659_NIB.png",
    "FileType": 4
  },
  {
    "CardholderID": 1,
    "FileContent": "/9j/4AAQSkZJRgABAQAAA//Z",
    "FileID": 2,
    "FileName": "images.jpg",
    "FileType": 4
  }
]
}
}
```

## 20. Update Profile API – *add files*

The following API endpoint enables program managers to submit additional cardholder documents for compliance processes.

Endpoint:

/api/Pro/UpdateProfile

Request:

```
"programId": 1,
"UpdateProfile1": {

  "CustomerId": 1,
  "CardholderID": "1558",
  "Profile": {},
  "IstFiles": [
    {
      "CardholderID": "1558",
      "FileContent": "/9j/4 GnHIWprRpytee5/9k=",
      "FileID": "1",
      "FileName": "99_passport.png",
      "FileType": "0"
    }
  ]
}
```

Notes:

- The CardholderID fields must have the same cardholder ID
- You can include as many documents as you need (minimum 2 the ID image and the proof of address mage.
- The max size for each document is 2 Megabytes.

## 21. Reissue Card (Sample)

Please see below an example of the reissue card method.

```
{
  "programId": 1,
  "ReissueCard1": {
    "ActivateNewCard_Setting": false,
    "CardReferenceId": Sample string1,
    "ReissueType": 3
  }
}
```

## 22. Compliance and Risk Field Values

Please see below the fields that should be used to send Compliance and Risk information.

**Customer Business Activity field was removed on 1/30/2023. Please make sure that it is not included in your system/request. Updated system to change the separator field from ^ to , (comma)**

Customer Subtype	Please use the field EmployerName to send the information	For example, if the customer subtype is "Employee with Fixed Salary" you should send a 6 in this field.
<del>Customer Business Activity</del>	<del>Please use the field EmployerAddress to send the information</del>	<i>This field was removed from the system and no longer needed</i>
Source of Funds	Please use the field EmployerTelephone to send the information	For example, if the source of funds is "Employment (Salaried)" you should send a 1133 in this field.
Customer Occupation	Please use the field Occupation to send the information	For example, if the source of funds is "Legal Occupations" you should send a 41 in this field.
ET_Amount_Incoming	Expected Transactional Activity External Incoming (Amount in USD\$) (Monthly)	Only for new DDA accounts, not user for prepaid cards.
ET_Amount_Outgoing	Expected Transactional Activity External Outgoing (Amount in USD\$) (Monthly)	Only for new DDA accounts, not user for prepaid cards.
ET_Quantity_Incoming	Expected Transactional Activity External Incoming (Quantity of Transactions) (Monthly)	Only for new DDA accounts, not user for prepaid cards.
ET_Quantity_Outgoing	Expected Transactional Activity External Outgoing (Quantity of Transactions) (Monthly)	Only for new DDA accounts, not user for prepaid cards.
FI_Account_Number	Financial institution Bank Account Number	Only for new DDA accounts, not user for prepaid cards.
FI_Financial_Institution	Name of Financial Institution	Only for new DDA accounts, not user for prepaid cards.
FI_Currency	Bank account currency	Only for new DDA accounts, not user for prepaid cards.

Fatca_US	Are you a US citizen, US resident or US taxpayer? (yes or NO)	Only for new DDA accounts, not user for prepaid cards.
Fatca_Tax_ID	If yes, please provide SSN	Only for new DDA accounts, not user for prepaid cards.
Fatca_Third_parties	Relationship with third parties	Only for new DDA accounts, not user for prepaid cards.
Source_of_Wealth		Only for new DDA accounts, not user for prepaid cards.
Account_Purpose		Only for new DDA accounts, not user for prepaid cards.
Account_Type		Only for new DDA accounts, not user for prepaid cards.
Product_Type		Only for new DDA accounts, not user for prepaid cards.

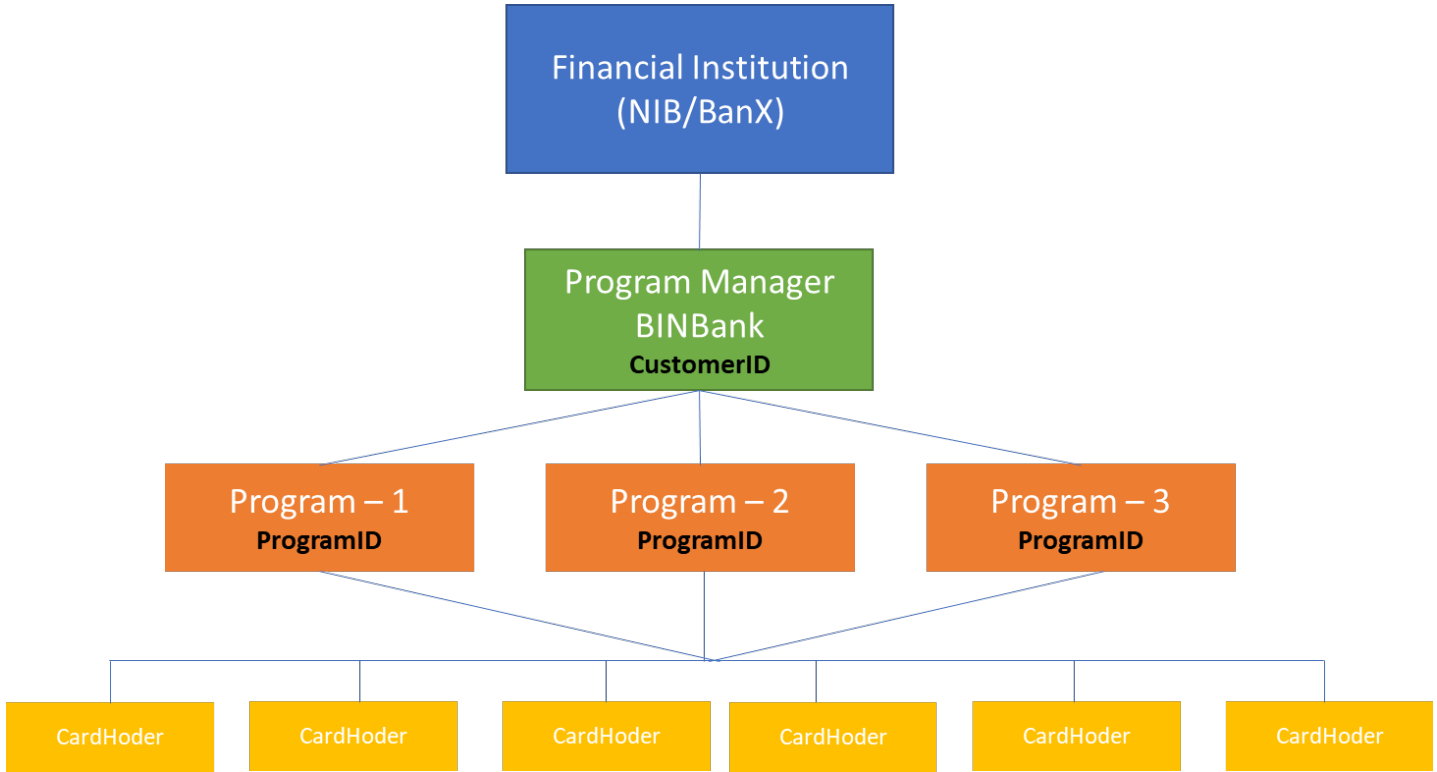
Please refer to the annex: BANTRANSFER API FILE Documentation v2.8 - Annex - 01.pdf to find the fields' valid values.

You can use the BANTRANSFER API FILE Documentation v2.9 - Annex – 01.XLSX to import the data or the file BANTRANSFER API FILE Documentation v2.9 - Annex – 01.txt (text file using the , “Comma” as a field separator)

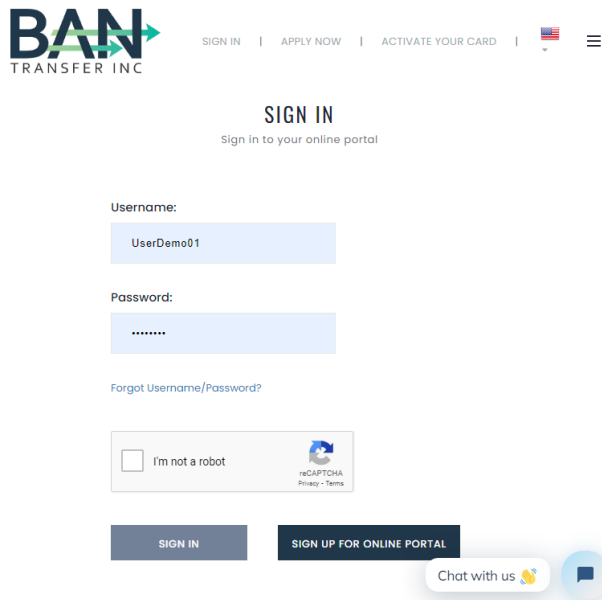
### 23. UpdateCardholderCredentials -

Purpose: Assign the username and password to a cardholder to give access to any channel (cardholder portal, mobile app, etc).

Entities Model.



Cardholder Portal





### 23.1 Endpoint.

/Cardholder/UpdateCardholderCredentials, this endpoint creates or updates the cardholder credentials

#### 23.1 Request.

```
{
  "customerId": 1,
  "cardholderId": 3454,
  "username": "JCTtest",
  "password": "1113"
}
```

#### 23.1 Response:

```
{
  "ExecutionResult": {
    "Exeption": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
  "Response": true
}
```

## 24. CardHolderAddCard-

To add a new card (second card) to an existing cardholder

### 24.1 Endpoint.

/Cardholder/CardHolderAddCard

This endpoint creates a new card to an existing cardholder

### 24.1 Request.

```
{
  "CustomerId": 1,
  "CardProgramId": 1,
  "cardholderId": 3430,
  "ReferenceId": "110195061867933"
}
```

Notes:

**cardholderId** Id of the cardholder the new card will be assigned to.

**Reference ID:** Id of the actual cardholder's card

### 24.1 Response:

```
{
  "BatchReferenceID": "- 110195061867933",
  "CardholderId": 3430,
  "ExecutionResult": {
    "Exeption": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
  "NameOnCard": null,
  "ProgramName": "_BanTransfer Prepaid Program",
  "TransId": "F2156338446",
  "TransactionCompleted": true
}
```

Notes:

**For additional cards the issuer processor DO NOT send back a batch ID, the field BatchReferenceID contains ONLY the separator ++ and the NEW card reference ID**

## 25. Sensitive Card information support

Purpose: Provide customers with the capability to use the BAN Transfer’s PCI environment to enable the cardholder to view / capture card sensitive information.

### 24.1 Display Card Information

#### 25.1 Display Card information.

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to display a card’s sensitive information (non-masked Card Number, expiration date, security code.

Sandbox endpoint: <http://34.227.56.76:8080>

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method (Please refer to Section 1 for any additional details).

Parameters: **(V1,V2,V3,V4, V13,V14)**

ID	Content	Example
V1	Card image / text box height	300
V2	Card image / text box width	400
V3	Card reference ID	12700001110141
V4	Card Holder ID	36578
V5	Text box font	“Lucida Console”
V6	Text box <b>font</b> size	8
V7	<b>Button Border</b>	<b>none</b>
V8	<b>Button color</b>	<b>white</b>
V9	<b>Button text align</b>	<b>center</b>
V10	<b>Button font size</b>	<b>16</b>
V11	<b>Button text decoration</b>	<b>none</b>
V12	URL for notification	https://mycard.help/response
V13	Request type	1 or empty: Display card 2: Capture new card PIN
V14	Customer Transaction ID	0acc58e5-4193-42a8-a758-71eb52cdb5ec

### 25.1 Capture new card pin.

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to **capture new card pin**.

Sandbox endpoint: <http://34.227.56.76:8080>

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API's method (Please refer to Section 1 for any additional details).

Parameters: **(V3,V4,V5,V6,V7,V8,V9,V10,V11,V12,V13,V14)**

ID	Content	Example
V1	Card image / text box height	300
V2	Card image / text box width	400
V3	Card reference ID	12700001110141
V4	Card Holder ID	36578
V5	Text box font	"Lucida Console"
V6	Text box <b>font</b> size	8
V7	<b>Button Border</b>	<b>none</b>
V8	<b>Button color</b>	<b>white</b>
V9	<b>Button text align</b>	<b>center</b>
V10	<b>Button font size</b>	<b>16</b>
V11	<b>Button text decoration</b>	<b>none</b>
V12	URL for notification	https://mycard.help/response
V13	Request type	1 or empty: Display card 2: Capture new card PIN
V14	Customer Transaction ID	51a07e20-fe27-4eea-b7b8-e76faef6ec8a

## 26. Bank Account Endpoints-

Purpose: Enable account holder to process account's related transactions

### 26.1. Create customer's Account.

To be included

### 26.2. Map account to cardholder (customer)

Not required, because to get an account the customer should have a card and then a cardholderID.

### 26.3. Get Core Banking CustomerID by CardholderID

End point: <https://qat.nibank.mobi/BusRestV2/api/pro/T24AccountsByCardholderID>

Request:

```
{
  "customerId": 1,
  "cardholderID": 1225
}
```

The cardholder ID that signed in on the mobile app or on the web portal

Response:

```
{
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
  "T24Accounts": {
    "T24AccountList": [
      {
        "T24AccountID": 8,
        "T24CustomerID": "100211",
        "T24customerMnemonic": "CEB002",
        "cardholderID": 1225
      }
    ]
  }
}
```

The important Field on the response is the T24CustomerID; this customer should be used to retrieve the customer's accounts from T24

## 26.4. Get Core Banking Accounts by Core Banking CustomerID

Endpoint: <https://qat.nibank.mobi/BusRestV2/api/pro/HoldingsAccounts>

On the request Customer ID, you should send the T24 mapped to the cardholderID

Request:

```
{
  "customerId": 1,
  "programId": 1,
  "Holdingaccounts": {
    "customerId": "100211"
  }
}
```

Response:

```
{
  "body": [
    {
      "IBAN": "AG8511110001300100000001",
      "accountId": "100009584",
      "availableBalance": "3.24",
      "availableFunds": "3.24",
      "availableLimit": "NOLIMIT",
      "clearedBalance": "3.24",
      "currencyId": "USD",
      "customerId": "100584",
      "customerName": "JOHN SMITH",
      "displayName": "JOHN SMITH",
      "onlineActualBalance": "3.24",
      "openingDate": "20191205",
      "productName": "White Account",
      "sortCode": "0001300",
      "workingBalance": "3.24"
    }
  ],
  "executionResult": {
    "Exception": "",
    "ReturnCode": "1000",
  }
}
```

```

        "ReturnDesc": "Success"
    },
    "header": {
        "audit": {
            "T24_time": 124,
            "requestParse_time": 50,
            "responseParse_time": 50,
            "versionNumber": null
        },
        "id": null,
        "page_size": 99,
        "page_start": 1,
        "page_token": "202301020720201210.01,99",
        "status": "success",
        "total_size": 1,
        "transactionStatus": null
    }
}

```

The `body.accountId` has the account ID. Please notice that a customer can have more than 1 account and then the body contains an array of accounts; in the above example the account number is: 100002442

## 26.5. Get Core Banking Account's transactions.

Endpoint: <https://qat.nibank.mobi/BusRestV2/api/pro/HoldingsTransactions>

Request:

```

{
    "customerId": 1,
    "programId": 1,
    "HoldingsTransactions": {
        "accountId": "100009584",
        "dateFrom": "2022-03-14T00:00:00",
        "dateTo": "2023-03-14T00:00:00"
    }
}

```

Response:

```
{
  "body": [
    {
      "bookingDate": "20220315",
      "debitCreditIndicator": "Debit",
      "transactionAmount": "2500",
      "transactionId": "BNK22074BK0BGFDH",
      "transactionName": "Prepaid Card Load Upik",
      "transactionTotal": "-2500",
      "valueDate": "20220315"
    },
    {
      "bookingDate": "20220315",
      "debitCreditIndicator": "Credit",
      "transactionAmount": "2500.00",
      "transactionId": "BNK22074BB0HMFDDG",
      "transactionName": "Payment Invoice A.Pk",
      "transactionTotal": "0",
      "valueDate": "20220315"
    },
    {
      "bookingDate": "20220315",
      "debitCreditIndicator": "Credit",
      "transactionAmount": "65.00",
      "transactionId": "BNK22074DDJBGJ0",
      "transactionName": "OTHER Commissions",
      "transactionTotal": "65",
      "valueDate": "20220315"
    },
    {
      "bookingDate": "20220331",
      "debitCreditIndicator": "Debit",
      "transactionAmount": "65",
      "transactionId": "AAACT22090GTRSCB7M",
      "transactionName": "Charge - Capitalise",
      "transactionTotal": "0",
      "valueDate": "20220331"
    }
  ]
}
```



```
    },
    {
      "bookingDate": "20220331",
      "debitCreditIndicator": "Credit",
      "transactionAmount": "0.01",
      "transactionId": "AAACT22090GTRSCB7M",
      "transactionName": "Interest - Capitalise",
      "transactionTotal": "0.01",
      "valueDate": "20220401"
    },
    {
      "bookingDate": "100002442",
      "debitCreditIndicator": null,
      "transactionAmount": null,
      "transactionId": "100211",
      "transactionName": "USD",
      "transactionTotal": null,
      "valueDate": "-57.63"
    }
  ],
  "executionResult": {
    "Exception": "",
    "ReturnCode": "1000",
    "ReturnDesc": "Success"
  },
  "header": {
    "audit": {
      "T24_time": 358,
      "requestParse_time": 61,
      "responseParse_time": 48,
      "versionNumber": null
    },
    "data": null,
    "page_size": 99,
    "page_start": 1,
    "page_token": "202301020720205078.01,99",
    "total_size": 5
  }
}
```

```
}  
}
```

## 26.6. Process a Core Banking Account to Card transfer.

Endpoint: <https://qat.nibank.mobi/BusRestV2/api/pro/OrderPaymentAtoC>

Request:

```
{  
  "customerId": 1,  
  "programId": 1,  
  "bankTransferCard": {  
    "AccountNumber": "100009584",  
    "Amount": 5.0,  
    "CardReferenceNumber": "110631059536",  
    "Comments": "Prueba",  
    "cardholderID": 1225  
  }  
}
```

Response:

```
{  
  "BankTransID": null,  
  "CardTransID": null,  
  "executionResult": {  
    "Exception": null,  
    "ReturnCode": "-1000",  
    "ReturnDesc": "Bank Transaction Error: Bad Request E-  
127331 - INPUT MISSING (E-113068), INPUT MISSING"  
  }  
}
```

## 26.7. Process a Card to Core Banking Account transfer.

Endpoint: <https://qat.nibank.mobi/BusRestV2/api/pro/OrderPaymentCtoA>

Request

```
{
  "customerId": 1,
  "programId": 1,
  "bankTransferCard": {
    "AccountNumber": "100002442",
    "Amount": 19.91,
    "CardReferenceNumber": "110631059536",
    "Comments": "Prueba",
    "cardholderID": 1225
  }
}
```

#### Response

```
{
  "BankTransID": null,
  "CardTransID": "F2158962140",
  "executionResult": {
    "Exception": null,
    "ReturnCode": "-400",
    "ReturnDesc": "Bad Request E-
119293 - DEBIT ACCOUNT IS MANDATORY"
  }
}
```

## 26.8. Get Core banking system account beneficiaries.

Endpoint: <https://qat.nibank.mobi/BusRestV2/api/pro/PartyBeneficiaries>

### Request

```
{
  "customerId": 1,
  "programId": 1,
  "PartyCustomers": {
    "customerId": "100130"
  }
}
```

### Response

```
{
  "body": [
    {
      "beneficiaryAccountId": "100001853",
      "beneficiaryId": "BEN2134800013",
      "customerReference": null,
      "dates": null,
      "nickName": "AGERE CAPITAL",
      "transactionName": null,
      "transactionType": null,
      "versionNumber": "1"
    },
    {
      "beneficiaryAccountId": "100000369",
      "beneficiaryId": "BEN2135500005",
      "customerReference": null,
      "dates": null,
      "nickName": "NIB CAPITAL & FINANCIAL SERVICES",
      "transactionName": null,
      "transactionType": null,
      "versionNumber": "1"
    }
  ],
  "executionResult": {
    "Exception": "",
    "ReturnCode": "1000",
    "ReturnDesc": "Success"
  },
  "header": {
    "audit": {
```

```
        "T24_time": 145,  
        "requestParse_time": 56,  
        "responseParse_time": 48,  
        "versionNumber": null  
    },  
    "id": null,  
    "page_size": 99,  
    "page_start": 1,  
    "page_token": "202301021134585853.01,99",  
    "status": "success",  
    "total_size": 8,  
    "transactionStatus": null  
}  
}
```

## 26.9. Process a Core Banking Account to Core Banking Account transfer.

Endpoint: <https://gat.nibank.mobi/BusRestV2/api/pro/OrderPaymentOrders>

Request

```
{
  "customerId": 1,
  "programId": 1,
  "Payload": {
    "body": {
      "PSDCompliant": null,
      "accountWithBankBIC": null,
      "accountWithBankClearingCode": null,
      "accountWithBankIBAN": null,
      "additionalInformations": [],
      "amount": 100.00,
      "beneficiaryAccountId": null,
      "beneficiaryAddresses": [],
      "beneficiaryBIC": null,
      ,
      "beneficiaryBankClearingCode": null,
      "beneficiaryCountryCode": null,
      "beneficiaryIBAN": null,
      "beneficiaryId": "BEN2134800013",
      "beneficiaryName": null,
      "chargeAccountCurrencyId": null,
      "chargeAccountId": null,
      "chargeBearer": null,
      "charges": [],
      "clearingChannel": null,
      "creditAccountIBAN": null,
      "creditAccountId": "100002302",
      "creditCurrencyId": null,
      "creditPortfolio": null,
      "creditValueDate": null,
      "debitAccountIBAN": null,
      "debitAccountId": "100000048",
      "debitAmount": 0,
      "debitCurrency": "USD",
      "debitValueDate": null,
      "endToEndReference": "7E5C10F9EDAB49ACA7AB005E0D409E95",
      "executionDate": null,
      "forexCustomerRate": null,
      "forexSpread": null,
      "indicativeRate": null,
      "instructionIdReference": null,
    }
  }
}
```

```

    "invoiceReferences": [],
    "localInstrumentCode": null,
    "narratives": [],
    "orderingBank": null,
    "orderingCustomerId": "100130",
    "orderingCustomerName": null,
    "orderingPortfolio": null,
    "orderingReference": null,
    "overrides": [],
    "paymentCurrencyId": "USD",
    "paymentOrderProductId": "ACTRFB",
    "purpose": null,
    "remittanceInformations": [],
    "requiredCreditValueDate": null,
    "signatories": [],
    "structuredCommunicationCode": null,
    "structuredCreditorReference": null,
    "structuredIssuer": null,
    "termsAndConditions": null,
    "totalDebitAmount": null,
    "transactionType": null,
    "treasuryRate": null
  },
  "header": {}
}
}

```

## 26.10.Account to Account process.

Please see below the step by step to process an account-to-account transaction.

1. Get/Retrieve the cardholder ID.
2. Execute **T24AccountsByCardholderID**. This endpoint will return the bank accounts that belong to the cardholder.
3. Execute the **PartyBeneficiaries** Get the bank account beneficiaries. In accordance with banking regulations, an account should have a registered beneficiary account to be able to process and account to account transactions.
4. Execute the **OrderPaymentOrders** to process the account-to-account transactions.
5. The body should have the below fields.

```

    "amount": 100.00,
    "beneficiaryAccountId": null,
    "beneficiaryId": "BEN2134800013",
    "chargeBearer": "OUR",
    "creditAccountId": "100002302",
    "debitAccountId": "100000048",
    "debitAmount": 0,

```

```
"debitCurrency": "USD",  
"endToEndReference": "7E5C10F9EDAB49ACA7AB005E0D409A12",  
"orderingBank": "sample string 34",  
"orderingCustomerId": "100130",  
"paymentCurrencyId": "USD",  
"paymentOrderProductId": "ACTRFB"
```



## 26.11.New Prospect

Please see below the request to register a new Customer - Prospect

```
{
  "customerId": 1,
  "programId": 1,
  "Prospect": {
    "body": {
      "customerNames": [
        {
          "customerName": "Mauricio Gomez"
        }
      ],
      "streets": [
        {
          "street": "Calle 16B 95-01"
        }
      ],
      "addresses": [
        {
          "address": "Casa 7"
        }
      ],
      "townCountries": [
        {
          "townCountry": "Bogota"
        }
      ],
      "postCodes": [
        {
          "postCode": "10001"
        }
      ],
      "countries": [
        {
          "country": "CO"
        }
      ],
      "phoneNumbers": [
        {
          "phoneNumber": "31595160981"
        },
        {
          "phoneNumber": "3102087741"
        }
      ],
      "emails": [
```

```
    {
      "email": "jc_mgomez@gmail.com"
    }
  ],
  "socialNetworkIds": [
    {
      "socialNetworkId": "mgomez"
    }
  ],
  "prospectLegalDetails": [
    {
      "legalId": "AB58771",
      "legalDocumentName": "Passport"
    },
    {
      "legalId": "792418541",
      "legalDocumentName": "National ID"
    }
  ],
  "personEntity": "PERSON",
  "gender": "MALE"
},
"header": {}
}
}
```

## 27. Display Card Information (iframe) - Sandbox

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to display a card’s sensitive information (non-masked Card Number, expiration date, security code.

Method: POST

Sandbox endpoint: [https://securebridge.360programsolutions.com/BridgeApi\\_360/api/Home/Index](https://securebridge.360programsolutions.com/BridgeApi_360/api/Home/Index)

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method.

Parameters: **(V1, V2, V3, V4, V13, V14)**

ID	Content	Example	
V1	Card image / text box height	300	
V2	Card image / text box width	400	
V3	Card reference ID	127008450568	
V4	Card Holder ID	36578	Coming soon
V13	Request type	1 or empty to Display card	
V14	Customer Transaction ID	0acc58e5-4193-42a8-a758-71eb52cdb5ec	Coming soon

To Test from Postman create a POST request with the following url

[https://securebridge.360programsolutions.com/BridgeApi\\_360/api/Home/Index?V1=300&V2=400&V3=127008450568&V13=1](https://securebridge.360programsolutions.com/BridgeApi_360/api/Home/Index?V1=300&V2=400&V3=127008450568&V13=1) and add a bearer token (authorization) with a token created with the verifyusercredentials endpoint.

On the response you will get an html with a URL to retrieve the card image.

Example:

```
<html>
```

The screenshot shows a REST client interface for a request to `https://securebridge.360programsolutions.com/BridgeApi_360/api/Home/Index?V1=300&V2=400&V3=12700845...`. The request method is `POST`. The authentication is set to `Bearer Token` with the token `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...`. The response is `200 OK` with a status of `2.64 s` and a size of `327 B`. The response body is shown in `Text` format as `<html>`.

## 28. Capture new pin card (iframe) - Sandbox

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to **capture new card pin**.

Method: Get

Sandbox endpoint: [https://apisandbox.nibank.com/Bridge\\_Api/api/Home/Index](https://apisandbox.nibank.com/Bridge_Api/api/Home/Index)

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method

Parameters: **(V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14)**

ID	Content	Example	
V3	Card reference ID	12700001110141	
V4	Card Holder ID	36578	
V5	Text box font	“Lucida Console”	
V6	Text box font size	8	
V7	Button Border	None	
V8	Button color	White	
V9	Button text align	Center	

V10	Button font size	16	
V11	Button text decoration	None	
V12	URL for notification	<a href="https://mycard.help/response">https://mycard.help/response</a>	
V13	Request type	2 for Capture new card PIN	
V14	Customer Transaction ID	51a07e20-fe27-4eea-b7b8-e76faef6ec8a	Coming soon

## 29. Display Card Information (iFrame) - Production

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to display a card’s sensitive information (non-masked Card Number, expiration date, security code.

Method: Post

Production endpoint: <https://secureapi.360programsolutions.com/BridgeApi/api/Home/Index>

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method.

Parameters: **(V1, V2, V3, V4, V13, V14)**

ID	Content	Example	
V1	Card image / text box height	300	
V2	Card image / text box width	400	
V3	Card reference ID	12700001110141	
V4	Card Holder ID	36578	Coming soon
V13	Request type	1 or empty to Display card	
V14	Customer Transaction ID	0acc58e5-4193-42a8-a758-71eb52cdb5ec	Coming soon

## 30. Capture new pin card (iFrame) - Production

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to **capture new card pin**.

Method: Get

Production endpoint: <https://secureapi.360programsolutions.com/BridgeApi/api/Home/Index>

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method

Parameters: **(V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14)**

ID	Content	Example	
V3	Card reference ID	12700001110141	
V4	Card Holder ID	36578	
V5	Text box font	“Lucida Console”	
V6	Text box font size	8	
V7	Button Border	None	
V8	Button color	White	
V9	Button text align	Center	
V10	Button font size	16	
V11	Button text decoration	None	
V12	URL for notification	<a href="https://mycard.help/response">https://mycard.help/response</a>	
V13	Request type	2 for Capture new card PIN	
V14	Customer Transaction ID	51a07e20-fe27-4eea-b7b8-e76faef6ec8a	Coming soon

### 31. Capture Credit Card Information to process a payment - Production

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to **capture and process a credit/debit card payment transaction**.

Method: Get

Production endpoint: <https://secureapi.360programsolutions.com/BridgeApi/api/Home/Index>

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API’s method

Parameters: (V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14)

ID	Content	Example	
V3	Payment Reference ID	12700001110141	Mandatory Card reference ID is required when the transaction intends to load a card with the amount of the processed payment.
V4	Card Holder ID	36578	Optional
V5	Text box font	“Lucida Console”	Mandatory

V6	Text box font size	8	Mandatory
V7	Text box border color	#156082	Color (in Hex)
V8	Button Border	None	Mandatory
V9	Button color	White	Mandatory
V10	Button text align	Center	Mandatory
V11	Button font size	16	Mandatory
V12	Button text decoration	None	Mandatory
V13	URL for notification	<a href="https://mycard.help/response">https://mycard.help/response</a>	Optional
V14	Request type	3 to process a credit card transaction	Mandatory
V15	Customer Transaction ID	51a07e20-fe27-4eea-b7b8-e76faef6ec8a	Mandatory
V16	Language	1	Mandatory 1 English 2 Spanish

Screen layout:

Name on Card\*

Card Number\*

Expiation (mmdd)\*

CVV\*

Zip Code\*

Amount\*

Email to send the receipt\*

## 32. Capture IMT Information to process a Send Money transaction - Sandbox

This feature enables any external application (web browser, mobile app with embedded browser) to send a request to **capture and process a International Money Transfer “send” transaction**.

Method: Get

Sandbox endpoint: [https://apisandbox.nibank.com/Bridge\\_Api/api/Home/Index](https://apisandbox.nibank.com/Bridge_Api/api/Home/Index)

Security: Authorization at header level: Bearer Token. Should be used in the same way it is used on any API's method

Parameters: (V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14)

ID	Content	Example	
V3	Card Reference ID	12700001110141	Mandatory Source of funds
V4	Card Holder ID	36578	Optional
V5	Text box font	“Lucida Console”	Mandatory
V6	Text box font size	8	Mandatory
V7	Text box border color	#156082	Color (in Hex)
V8	Button Border	None	Mandatory
V9	Button color	White	Mandatory
V10	Button text align	Center	Mandatory
V11	Button font size	16	Mandatory
V12	Button text decoration	None	Mandatory
V13	URL for notification	<a href="https://mycard.help/response">https://mycard.help/response</a>	Optional
V14	Request type	4 when receive method is: Direct to Bank Account  5 when receive method is: Cash Pickup	Mandatory
V15	Customer Transaction ID	51a07e20-fe27-4eea-b7b8- e76faef6ec8a  The ID must be unique	Mandatory
V16	Language	1	Mandatory  1 English 2 Spanish



Screen layout:

Select a bank\*

**Receiver Information**

ID first name\*

ID middle name

ID last name\*

ID second last name

Account number\*

Confirm Account number

Send Amount\*

**Confirm**

Receive Amount

Receiver Country Change

Notes

**Process Transaction**

### 33. Reload Card – ATM Cash In

To allow cardholders to reload their card using the ATM this endpoint generates the one-time token required.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/ReloadCardTokenCreate>

Request:

```
{
  "customerId": 1,
  "ReloadCardTokenCreate1": {
    "LoaderType": "Atrium",
    "MonedasId": "USD",
    "NCRCodigoRemitenteId": 1,
    "ProgramId": 3,
    "RemesaRemitId": 949962,
    "Token_Amount": 100,
    "Token_CardReferenceID": "sample string 6",
    "Token_PhoneNumber": "sample string 7"
  }
}
```

Notes:

Nccodigoremitenteid will always be 1.

LoaderType will always be Atrium.

MonedasId is the Currency ISO code

Token\_Amount will be the amount that you want to cash in. Must be an integer without any decimal separator or decimal places

Token\_CardReferenceID will be the card reference id for the card being reloaded.

Token\_PhoneNumber you can add your phone number in this field, this is an optional field.

RemesaRemitId is always 949962 in Sandbox. Will be different in Production

### 34. Reload Card – ATM Locator

To allow cardholders to locate the ATM that can be used for the cash in (please refer to previous section)

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/GetAtmLocation>

Request:

```
{
  "customerId": 1,
  "AtriumPSNCRGetATMLocations": {
```

```

    "IntTransactionamount": 100,
    "Longitud": "-80.335112",
    "Latitud": "25.817635",
    "TipoTransaccion": 1
  }
}

```

**TipoTransaccion:** will always be 1

## 35. Reload Card – Barcode Cash In

To allow cardholders to reload their card at a store this endpoint is used to generate the one-time barcode to be shown.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/ReloadCardCreateTokenBarcode>

Request:

```

{
  "customerId": 1,
  "AtriumPSPSPaymentPostInput": {
    "LoaderType": 1,
    "ProgramId": 1,
    "Token_CardReferenceID": "sample string 3",
    "Token_PhoneNumber": "sample string 4",
    "amount": 5.0,
    "customerId": "sample string 6",
    "customer_takeover_data": {
      "address1": "sample string 1",
      "city": "sample string 2",
      "country_iso2": "sample string 3",
      "date_of_birth": "sample string 4",
      "email": "sample string 5",
      "first_name": "sample string 6",
      "last_name": "sample string 7",
      "phone_number": "sample string 8",
      "postcode": "sample string 9"
    }
  }
}

```

## 36. IMT Get Quote

To receive a quote for an International Money Transfer transaction, please use the following endpoint.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/IMTQuote2024Q3>

## Request:

```
{
  "programId": 1,
  "IMTQuoteInput": {
    "Amount": 100,
    "CityID": 0,
    "CountryID": "484",
    "CurrencyID": "MP",
    "SenderReferenceId": "127007806320",
    "paymentMethod": "1"
  }
}
```

**Amount:** The amount to send to the foreign country. E.g. 100, 200. 100.0 or 100.00 is an invalid amount. Must be an integer without any decimal separator or decimal places.

**CountryID:** ISO 3166 alpha-3 code of the destination country (e.g. 484 for Mexico, 170 for Colombia, 320 for Guatemala)

**CurrencyID:** ISO 4217 alpha-3 code of the destination currency. (e.g. MXN for Mexican pesos, COP for Colombian Pesos, GTQ for Guatemala Quetzales).

**SenderReferenceId:** Card reference ID of the cardholder that is initiating the IMT transaction.

**paymentMethod:** 1 for cash pick up, 2 for deposit in bank account

## Response:

```
{
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
  "Response": {
    "Amount": 100.0,
    "Amount_to_Send": 1751.00,
    "Comments": null,
    "Commission": 8.0,
    "Destination_Currency": "MP",
    "ExchangeRate": 17.51,
    "QuoteID": "b2e13e2b-38fb-4106-9d34-ad6bd4df99af",
    "Response": {
```

```
"Code": "1",
"Description": "b2e13e2b-38fb-4106-9d34-ad6bd4df99af^Quote Calculated"
},
"SDTOficinas": [
  {
    "OficinasId": "37128",
    "OficinasNombre": "BANCOPPEL - ACOLMAN-VICENTE VILLADA SN . COLONIA LOS PINOS"
  },
  {
    "OficinasId": "37129",
    "OficinasNombre": "BANCOPPEL - ATZACOALCO-AV. CENTENARIO 2144 SANTIAGO ATZACOALCO"
  },
  {
    "OficinasId": "37185",
    "OficinasNombre": "BANCOPPEL - CALIMAYA-CALLE JUAREZ 18 - COLONIA CENTRO"
  },
  {
    "OficinasId": "69053",
    "OficinasNombre": "INTERMEX (New CH0288)-CHEDRAUI - MONTECITO 38 NAPOLES"
  },
  {
    "OficinasId": "69572",
    "OficinasNombre": "INTERMEX (New CH0288)-CHEDRAUI - MONTECITO 38 NAPOLES"
  },
  {
    "OficinasId": "46895",
    "OficinasNombre": "OXXO-AGUSTIN REZA MEX-AV TLAHUAC 6701"
  },
  {
    "OficinasId": "45798",
    "OficinasNombre": "OXXO-AHORRO MEX-AHORRO POSTAL 61"
  },
  {
    "OficinasId": "45825",
    "OficinasNombre": "OXXO-AHUALTEPEC MEX-TEPETL L41"
  }
]
```

```

    }
  ],
  "Source_Currency": "USD",
  "Total_to_Debit": 108.0
}
}

```

## 37. IMT Transaction

To initiate an International Money Transfer transaction, please use the following endpoint.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/IMTTransaction>

Request:

```

{
  "SenderReferenceId": "127007806320",
  "IMT_Receiver Receiver": {
    "FirstName": "Camilo",
    "MiddleName": "Andres",
    "LastName": "Perez",
    "SecondLastName": "Martinez",
    "NationalID": "798471511",
    "NationalIDCountryID": "484",
    "Address": "Calle Real # 12-15",
    "CountryID": "484",
    "StateID": "DF",
    "CityID": "Ciudad de Mexico",
    "PhoneNumer": "5248741488551",
    "Bank": "BANORTE",
    "AccountNumber": "01054906",
    "AccountType": "1"
  },
  "Amount": "100.00",
  "Comments": "My first transaction",
  "QuoteID": "5418FB8C-EFF9-44E9-8620-6754B4A1EDB4"
}

```

**QuoteID:** This field must contain a quote ID received on the response of IMTQuote endpoint (please refer to previous endpoint section)

## 38. IMT Get Cities

To retrieve a list of the cities where the IMT can be sent.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/IMTGetCities>

Request:

```
{
  "programId": 1,
  "customerId": 2,
  "IMTCitiesInput": {
    "CompaniaId": 1,
    "MonedasIso": "sample string 2",
    "OficinasModoPagoNum": 3,
    "PaisesIsoAlfa3": "sample string 4"
  }
}
```

**CompaniaId:** will always be 1

**MonedasIso:** ISO 4217 alpha-3 code of the destination currency (MXN-Mexican Peso COP-Colombia Peso USD-dollar GTQ-Quetzal USD-El Salvador)

**OficinasModoPagoNum:** 1 for cash out 2 for bank deposit

**PaisesIsoAlfa3:** ISO 3166 alpha-3 code of the destination country (e.g. 484 for Mexico, 170 for Colombia, 320 for Guatemala)

## 39. IMT Get Offices

To retrieve a list of the location where the IMT can be sent for cash pickup.

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/IMTGetOffices>

Request:

```
{
  "programId": 1,
  "customerId": 2,
  "IMTOfficesInput": {
    "CiudadesId": "sample string 1",
    "MonedasId": "sample string 2",
    "OficinasModoPagoNum": 1,
    "PaisesId": "sample string 3",
    "ZonaId": 5
  }
}
```

**CiudadesId:** Obtained in the response for endpoint GetCities (please refer to previous section)

**MonedasId:** ISO 4217 alpha-3 code of the destination currency (MXN-Mexican Peso COP-Colombia Peso USD-dollar GTQ-Quetzal USD-El Salvador)

**OficinasModoPagoNum:** will always be 1

**PaisesId:** ISO 3166 alpha-3 code of the destination country (e.g. 484 for Mexico, 170 for Colombia, 320 for Guatemala)

**Zonald**

## 40. IMT Cancel Transaction

To cancel an IMT Transaction

Method: Post

URL: <http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/IMTCancelTransaction>

Request:

```
{
  "programId": 1,
  "customerId": 2,
  "IMTCancelTransactionInput": {
    "RemGiroRefAux": "sample string 1"
  }
}
```

Response:

```
{
  "APICancelarTransaccionResponse": {
    "SDT_DataCancelEstatus": "sample string 1",
    "SDT_DataCancelMensaje": "sample string 2",
    "SDT_DataCancelRefAuxRem": "sample string 3"
  },
  "ExecutionResult": {
    "Exeption": "sample string 1",
    "ReturnCode": "sample string 2",
    "ReturnDesc": "sample string 3"
  }
}
```

## 41. Program Owner (PO) Webhook

The PO will receive a notification from BANTransfer using the webhook we've implemented every time a new card is issued in the PO program. Following are the details:

Method: Get

URL: <https://ProgramOwnerDomain/webhook/BanxData>



Header:

Key: **x-api-key**

Value: **an API Key assigned by the PO**

Body:

```
{
  "cardreferenceID": "127007786522",
  "cardholderID": 45784,
  "cardholderForeingID": "79241654",
  "cardholderEmail": "name@domain.com",
  "cardholderDOB": "MM/dd/yyyy",
  "cardholderFirstName": "Mario",
  "cardholderMiddleName": "Enrique",
  "cardholderLastName": "Perez",
  "cardholderSecondLastName": "Gomez"
  "comments": "sample string 3"
}
```

This webhook will accept the above body and match authorization code and amount with the database record. If record matches, then webhook marks that token as used

## 42. Map Cards by Card ReferenceID

To add/map an additional card to a cardholder using the Card Reference ID.

Method: Post

URL: [http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/MAP\\_CardsbyReferece2024Q3](http://gat.nibank.mobi/BusRestLimited2025Q1V0/api/pro/MAP_CardsbyReferece2024Q3)

Request:

```
{
  "programId": 1,
  "CustomerID": 2,
  "MAP_CardsbyReferece": {
    "NewReferenceId": "sample string 1",
    "OriginalReferenceId": "sample string 2"
  }
}
```

NewReferenceId: The card reference ID that will be linked to the existing card.

OriginalReferenceId: The original/existing card's reference ID that will be linked to the new card.

Response:

```
{
  "CardNumber": "127008381743",
  "CustomerID": 1,
  "CxrefID": 3071,
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  }
}
```

CxrefID: ID assigned to the mapped accounts relation.

If return code = -100 means the original reference id is invalid

If return code = -200 means the new reference id is invalid

### 43. Request One Time Pin (OTP)

To request an OTP from the system for the user.

Method: Post

URL: <https://gat.nibank.mobi/BusRestLimited2025Q1V0/api/Cardholder/RequestOTP>

Request:

```
{
  "customerId": 1,
  "cardholderId": 4171,
  "programId": 1,
  "sendtype": "SMS",
  "emailOrPhone": "13054326653"
}
```

cardholderId: The existing cardholder ID in the system.

sendtype: The method in which the OTP will be sent will always be SMS.

emailorPhone: the phone number that will be receiving the OTP

Response:

```
{
  "ExecutionResult": {
```

```
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
  "Response": {
    "CardholderId": 4171,
    "CardholderName": null,
    "Expiration": 15,
    "OTPCode": "664169",
    "OTPTType": 4,
    "ProgramName": "__Test Program(pgm id 1)"
  }
}
```

Expiration: Time in minutes that the OTP is valid.

OTPCode: One time pin for the user

## 44. Validate One Time Pin (OTP)

To validate that the OTP that was used is valid in the system.

Method: Post

URL: <https://qat.nibank.mobi/BusRestLimited2025Q1V0/api/Cardholder/ValidateOTP>

### Request:

```
{
  "customerId": 1,
  "cardholderId": 4171,
  "OTP": "664169"
}
```

cardholderId: The card reference ID that will be linked to the existing card.

OTP: The OTP that was input by the user.

### Response (if the OTP is valid):

```
{
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "Success."
  },
}
```

```
    "Response": true
}
```

**Response (if the OTP is invalid or expired):**

```
{
  "ExecutionResult": {
    "Exception": null,
    "ReturnCode": "1000",
    "ReturnDesc": "OTP code not valid or expired."
  },
  "Response": false
}
```

## 45. Error Codes

Please see below the list of processor's error codes.

Code	Description	Abbreviation
DG	Direct Deposit Account Info not found	DirDeplInfoNotFo
0	Approved or Completed Successfully	Approved
1	Refer to Card Issuer	Ref2CardIssuer
2	Refer to Card Issuer's Special Conditions.g	Ref2ClssSpCng
3	Invalid Merchant	InvalidMerchant
4	Pick-Upl	Pick-upl
5	DO Not Honor	Do Not Honor
6	Error	Error
7	Pick-up Card, Special Condition	PickCardSpCnd
8	Honor With Identification	HonorWthldntif
9	Request In Progress	Request In Prog
11	Approved (VIP)	Approved (VIP)
12	Invalid Transaction	Inv Transaction
13	Invalid Amount	Invalid Amount
14	Invalid Card Number (No Such Number)	Inv Card No.
15	No Such Issuer	No such Issuer
16	Approved, Update Track 3	App,Upd Trek 3
19	Re-Enter Transaction	ReEntTransact
20	Invalid Response	InvalidResponse
23	Unacceptable Transaction Fee	UnacptTrnscFee
24	File Update Not Supported By Receiver (Only In 03xx)	ins rer,03xx su
25	Unable to Locate Record On File (Only In 03xx)	I in 03xx i of
26	Duplicate File Update Record, No Action (Only lin 03xx)	DpFIUpdRecNA
27	File Update Field Edit Error (Only In 03xx)	FUFEE oy i 03xx
28	File Update Record Locked Out (only in 03xx)	FURLO oy i 03xx
29	File Update Not Successful, Contact Acquirer (Only in 03xx)	FIUpdNtSucCnAc
30	Format Error	Format Error
31	Bank Not Supported By Switch	BankNtSpBYSwc
33	Expired Card	Expired Card
34	Suspected Fraud	Suspected Fraud
35	Card Acceptor Contact Acquirer, Pick-Up	CAcpCntAcq,Pick
36	Restricted Card	Restricted Card
37	Card Acceptor Call Acquirer Security	CdAcpCIIAcqSec
39	No Credit Account	NoCredAccnt
40	Requested Function Not Supported	ReqFuncNtSup
41	Lost Card	Lost Card
42	No Universal Account	NoUnivAccnt
43	Stolen Card, Pick-Up	StlnCrd,PickUp
44	No Investment Account	NoInvstmntAcnt
51	Not Sufficient Funds	NSF

52	No Checking Account	NoCheckngAcct
53	No Savings Account	NoSavngAccnts
54	Expired Card	Expired Card
56	No Card Record	No Card Record
57	Transaction Not Permitted to Cardholder	TrnsNtPerm2CH
58	Transaction Not Permitted	TransNotPerm
59	Suspected Fraud	Suspected Fraud
DE	Duplicate Record found against SSN/DL/Foreign Ids	DupSSNFailed
60	Card acceptor Contact Acquirer	CdAccpCntAcq
61	Exceeds Amount I Frequency Limit(s)	ExcdsWthdrwLmt
62	Restricted Card	Restricted Card
64	Original Amount Incorrect, Reversal (Unsupported)	OAIR USupported
65	Exceeds Withdrawal Frequency Limit	ExcdWthdFreqLmt
66	Card Acceptor Call Acquirer's Security	CrdAcpcIAcqSec
67	Hard Capture (Pick-Up)	HrdCapt(PickUp)
7	special conditions	Pick-up cardujj
76	Key Synchronization Error (eFunds Corporation)	KySynEr(EfdCrp)
77	Reserved For Private Use	ResForPrvtUse
78	Customer Not Eligible For POS (StarSM)	CstNtElgbFrPOS
79	Invalid Digital Signature	InvDgtSignature
80	Stale Dated Transaction (StarSM)	StaleDtdTrnsct
81	Issuer Requested Standin	IssReqStndIn
82	Count Exceeds Limit (VISANet)	Cnt Excds lmt V
83	Reserved For Private Use	Resr PVT use
84	Time Limit For Pre-Authorization Reached (VISANet)	Tim Lmt P-Auth
85	Not Decline (Valid only AVS Only or ecommerce certification requests)	Not Decline
87	Check Already Posted	ChckAlrdyPostd
88	Information Not On File	InfNtonFile
89	Card Verification Value (CVV) Verification Failed (No PickUp)	cvv FailNoPik
90	Cutoff In Progress	CutoffInProgrs
91	Issuer Or Switch Inoperative	Iss/Swtch InOp
92	Financial Institution Or Intermediate Network Unknown For Routing	FncIns/InNwkUn
93	Transaction Cannot Be Completed Due to A Violation Of The Law	TrncNtPmlwViol
94	Duplicate Transmission	DupTransmiss
95	Reconcile Error	Reconcile Error
96	Refer to Card Issuer	Ref2CardIssuer
NO	Unable to Authorize	Unable to autho
NI	Authorization Life Cycle Expired	exp auth Life C
R2	Allowable PAN Entries Warning-Approved	Allo Pan War -a
so	Check Not Acceptable For Cash	ch !accpable ch
SI	Check Not Acceptable	ch !accpable
S2	Check Deposit Limit Exceeded	ch dep lmt exd
S3	Cash Back Limit Exceeded	ch back lmt exc

S4	Check Amount Does Not Match Courtesy Amount	chk amt !mat CA
S7	Unmatched Voucher Information	!mat vouch info
S8	Allowable PAN Entries Exceeded -Denial	-Allow Pan Exce
S9	Expiration Date Match	exp dte !match
SA	Inactive Card	Inactive card
SB	Expiration Date Mismatch (Card Pickup)	Exp Date !match
SC	Item Suspected For Stop Pay	itm for stop py
SD	Account Closed	Acct closed
SE	Ineligible Account	!eligibke acct
SF	Item Submitted More Than Two Times	submtd>itm *2
SG	No Account On File - Absolute	! Acct on file
SH	Unable to Locate	can't locat
SI	General Denial	Gen denial
SJ	Item Settled Via ACH	itm sttd ACH
SK	Cross-Reference Card Not Found	!Fnd C-Ref card
SL	Category Limit Exceeded	>CategoryLmt
SM	Transaction Limit Exceeded	>Trans Lmt
SN	Daily Limit Exceeded	>Daily lmt
so	Monthly Limit Exceeded	>Mon lmt
SP	Invalid Secret Code	secre cd inv
SR	Bad CVV2	bad cw2
10	Approved For Partial Amount	App4PartAmount
LS	Cannot link card due to supplementary card	Cant link/SpCrd
LV	Limit Parameter Violation	Invalid Drv Lie
PT	Phone number not exist	Phone Not Exist
MS	Maximum allowed supplementary cards limit exceeds.	MaxSuplCardlmt
2Z	Reserved For ISO Use	ResForISOUse
3Z	Reserved For ISO Use	ResForISOUse
4Z	Reserved For ISO Use	ResForISOUse
5Z	Reserved For ISO Use	ResForISOUse
6Z	Reserved For ISO Use	ResForISOUse
7Z	Reserved For ISO Use	ResForISOUse
8Z	Reserved For ISO Use	ResForISOUse
gz	Reserved For ISO Use	ResForISOUse
E1	Account number invalid	ACCTLENGTHERR
E2	Verification error	CHECK DIGIT ERR
E3	Verification error	CID FORMAT ERRO
E4	Error(96) see MRC response	ERROR
E5	Cash back service not available	CASHBACK NOT AV
E6	Invalid expiration date	DECLINE
E7	CVV2 value supplied is invalid	CVV2 MISMATCH
ES	Re-Submit	TIME OUT
E9	Invalid acquirer	Inv Acquirer
EA	Acquirer user ID or password is not correct	Acq Bad User
EB	Wrong account access code	WRONGAAC
EC	Wrong account number specified for the card	WRONGACC NO

ED	Reversal failure	REV FAILED
EE	Reverse charge action failed	REV CHRG FAILED
EF	Invalid ACH account number	INVALID ACHACCN
EG	Successful ACH verification	ACH VERF SUCCES
EH	Unsuccessful ACH verification	ACH VERF FAILED
EI	Failed to withdraw amount from card or bank account	WITHDRAW FAILUR
EJ	Failed to load amount to card or bank account	LOAD FAILURE
EO	Client System Malfunction	Client Sys Err
ER	Error ? see MRC response	ERROR
MI	Already Active	Card Already Ac
TB	Account Number is Either not registered or not Found	Getting ACH Inf
T4	Exceeds maximum load amount per day week	EXCEEDS MAX PER
QO	Invalid date	DATE ERROR
T7	Destination is not found	INVALID ROUTING
QI	Invalid expiration date	DECLINE
IP	Invalid Payee	Invalid Payee
	Card activation is required to register, go back and choose	
PA	activate.	PreAct
IL	Invalid Card to Card Transfer Parameter Limit	Invlimit
UC	Card Can Not Be Unlinked	CardCanNotUnlk
ML	Maximum allowed linked card limit exceeded	MaxLnkCrdlimit
AL	Card is Already Linked	CrdAlrdylinked
LD	Card Linking Denied	CrdLnkingDnied
IB	Invalid Card BIN/Invalid Reference ID	InvBIN
ID	Instance is not allowed to acquirer	InstDenied
ST	No service exist	SrvExt
UN	User does not exist for the provided Card Number	NoUsr
RI	Invalid Bank Routing Number	InvRoutNum
CT	Card Track already generated	CardTrkGen
OF	OFAC failed	OFAC failed
AD	Activation Denied due to Imminent Card Expiry	ActDenDuelImmExp
FI	Fixed Initial Value Card, No Load Amount Required	FixInItValCard
IA	Inactive Acquirer	InactiveAcquire
AS	Unable to add stakeholder record	UnableAddStkhol
us	Unable to update stakeholder record	UnableUpdStkhol
DS	Unable to delete stakeholder record	UnableDelStkhol
IC	Invalid Card Program	InvCardPrg
ss	Stakeholder does not exist in sales chain or invalid sales node number	InvStkhldrSalNo
IR	Invalid Card Range	InvRange
AR	Unable to assign/reassign card	FailAssignReAss
UR	Unable to unassign	FailUnAssign
CD	Invalid Card Design	InvCardDesign
CL	Invalid Card Logo	InvCardLogo
oz	Reserved For ISO Use	ResForISOUse



IZ	Reserved For ISO Use	ResForISOUse
FB	Card is Fraud-Blocked	FraudBlock
IM	Invalid Member Number	InvalidMemberNo
MA	Aggregate sum of loads within last 30 days exceeds limit	MonthlyLodAmt
MC	Number of loads within last 30 days exceeds limit	MonthlyLodFrq
DA	Aggregate sum of loads within last 24 hours exceeds limit	DailyLodAmt
CB	Card balance exceeds limit	CardBalLimit
NA	Payment Processor not available	PROC NOT AVL
GA	Unsupported character found in the request	UnSupportedChar
HE	Problem with HSM Communication	HSM Error
DD	Data definition issue.	DataDefinitionI
OD	Returned per ODFI's request	Returned per OD
AO	Authorization Revoked by Customer(adjustment entries)	Authorization R
PS	Payment Stopped	PaymentStop
CA	Customer advises not authorized	AdvisesNotAuth
UF	uncollected Funds	uncollected Fun
NR	Non-participating RDFI	RDFI Not Qualif
NT	Non-Transaction Account	Non-Transaction
ZP	Invalid ZIP/Postal Code	InvalidZipPostI
DO	Invalid Date of Birth	Invalid DoB
NS	Invalid SSN (Social Security Number)	Invalid SSN
PH	Invalid Phone Number	Invalid PhoneNo
DL	Invalid Driving License	Invalid Drv Lie
AV	AVS failed	AVS failed
BI	Invalid Bank Information Supplied	InvBankInfo
97	Already Activated Card	CrdAlrdyActv
98	Reserved For National Use	Res4NationIUse
99	Reserved For National Use	Res4NationIUse
45	Reserved For ISO Use	ResForISOUse
46	Reserved For ISO Use	ResForISOUse
47	Reserved For ISO Use	ResForISOUse
48	Reserved For ISO Use	ResForISOUse
49	Reserved For ISO Use	ResvdForISOUs
50	Reserved For ISO Use	ResForISOUse
69	Reserve For Future Use	ResvForFutUse
70	Reserve For Future Use	ResForFutUse
71	Reserve For Future Use	ResForFutUse
72	Reserve For Future Use	ResForFutUse
73	Reserve For Future Use	ResForFutUse
74	Reserve For Future Use	ResForFutUser
N2	Non-Receipt Of Requested Item (Future)	!rec reqd item
N3	Non-Receipt Of Requested Item, Illegible Copy (Future)	! recReq it II
N4	Transaction Exceeds Floor Limit (Future)	trans exds fl I
N5	Declined Authorization	Decline Auth
N6	Non-Matching Account Numbers (Future)	! Account numbe
N7	Error In Addition (Future)	Err in Addition

NS	Altered Amount (Future)	alt amt (future
N9	Incorrect Account Number (Future)	inc acct number
PO	Missing Signature (Future)	miss signature
PI	Slip Without Card Imprint (Future)	Slip WO Card
P2	Imprinting Of Multiple Slips (Future)	im multi sips
P3	Canceled Pre-Authorization Transaction (Future)	Can pre-auth tr
P4	Delinquent Settlement (Future)	Del settlement
P5	Currency Conversion Error (Future)	Curey conv err
P6	Credit Posted As A Debit (Sale) (Future)	cred postd deb
P7	Claim Or Defense (Future)	Claim/ defense
PS	Non-Receipt Of Goods (Future)	! recpt of good
pg	Defective Merchandise (Future)	def merchandise
RO	Fraudulent Transaction Prior to Embossed Valid Date (Future)	FT pr to embd V
RI	Credit Not Received (Future)	Cred ! received
17	Customer Cancellation, Reversal (unsupported)	CustCanc,Rev
18	Customer Dispute, Chargeback (Future)	CustDispChgBck
21	No Action Taken, Reversal (Unsupported)	NoActTak,Rev
22	Suspected Malfunction	SuspctMailFunc
68	Response Received too Late, Reversal (Unsupported)	Response receiv
32	Completed Partially	CompPartially
63	Security Violation(May Also Be A Chargeback)	SecViol,mayChrg
AT	Authorization code is required	AuthCodReq
RL	The referrer or Relay response URL is invalid	RefResplInvalid
TI	The Transaction ID is invalid	TransInvalid
TN	The Transaction was not found	TransNotFond
CN	The card type is not accepted	RestCrdTyp
CF	Configurations are invalid	PrcsCnfg
RQ	Mandatory field required	ReqFld