



# SOFTY COMP API

# IMPLEMENTATION GUIDE

**REVISION HISTORY**

Date	Version	Summary of Changes	Prepared by/ Changed By
2019-07-01	2.0	Documentation Creation	Anél Rosewall
2019-07-04	2.1	Changes the response “message” field for Client and Collection Insert/Update methods. Change type from string to array.	Anél Rosewall
2019-07-04	2.2	Added Mandate field to collections and added the mandate download section.	Anél Rosewall
2019-07-11	2.3	Added AVS request functionality	Anél Rosewall
2019-07-22	2.4	Added Filtering to the client list to find clients faster	Anél Rosewall
2019-07-25	2.5	Added Avs Client Implementation guide	Anél Rosewall
2019-07-26	2.6	Added Avs Client Example project	Anél Rosewall
2019-09-16	2.7	<ul style="list-style-type: none"> <li>• Add CDV Check</li> <li>• Add Product List, Insert and Update</li> <li>• Add SAPI Documentation Download Functionality</li> </ul>	Anél Rosewall
2020-06-17	2.8	<ul style="list-style-type: none"> <li>• Add Client History Details</li> <li>• Add Collection History Details</li> <li>• Add List Collection Filter</li> </ul>	Kevin Coetzee
2020-06-26	2.9	<ul style="list-style-type: none"> <li>• Add List All Collections</li> <li>• Made the “GET UNPAIDS BY COLLECTION” response time faster.</li> </ul>	Kevin Coetzee
2020-10-13	3.0	<ul style="list-style-type: none"> <li>• Add List Transactions by Date Range</li> <li>• Add list Unpaid Transactions by Date Range</li> </ul>	Kevin Coetzee
2021-02-10	3.1	<ul style="list-style-type: none"> <li>• Add Reference Field to Products Insert and Update methods.</li> </ul>	Kevin Coetzee
2021-02-23	3.2	<ul style="list-style-type: none"> <li>• Add DebiCheck Collection</li> <li>• Amend DebiCheck Collection</li> <li>• Cancel DebiCheck Collection</li> <li>• Get DebiCheck Collection Details</li> </ul>	Hennie Kleynhans

## TABLE OF CONTENTS

Revision History.....	2
Introduction .....	7
Latest Api Documentation .....	7
API Fields and Rules.....	7
Fields .....	7
Authentication Operations .....	8
Authentication User Sequence Diagram .....	8
Authentication User request Fields .....	8
Sample Code for Authentication Request.....	8
Authentication Response Fields .....	9
Authedication Response Sample .....	9
Client Operations.....	9
Get Client List .....	9
Client Fields .....	9
Client List Filtering Request Fields .....	10
Request Sample .....	11
Response Sample.....	11
Get Client Details.....	12
Request Sample .....	12
Response Sample.....	12
Get Client History Details.....	13
Request Sample .....	13
Response Sample.....	13
Insert Client .....	14
Request Sample .....	14
Response Fields .....	14
Response Samples .....	15
Update Client.....	15
Request Sample .....	15
Response Fields .....	16
Response Sample.....	16
Collection Operations .....	17
Get Collection List.....	17

Collection Fields.....	17
Request Sample .....	18
Response Sample.....	18
Get Collection Filter List.....	19
Collection Fields.....	19
Request Sample .....	21
Response Sample.....	21
Get ALL Collection List .....	21
Collection Fields.....	22
Request Sample .....	23
Response Sample.....	23
Get Collection Details .....	24
Request Sample .....	24
Response Sample.....	24
Get Collection History Details .....	25
Request Sample .....	25
Response Sample.....	25
Create Collection .....	26
Request Sample .....	26
Response Fields .....	26
Response Samples .....	27
Update Collection.....	27
Request Sample .....	27
Response Fields .....	28
Response Samples .....	28
DebiCheck Collections .....	29
DebiCheck Collection Fields .....	29
Table Types and Definitions .....	33
Entry Class Codes Types.....	33
Frequency Types.....	34
Mandate Status Types .....	34
Operation Types .....	34
Mandate Amendment Reasons.....	34
Mandate Cancellation Reasons .....	34
Create DebiCheck Collection.....	35

Request Sample .....	35
Response Fields .....	36
Response Samples .....	37
Update DebiCheck Collection.....	37
Request Sample .....	37
Response Fields .....	39
Response Samples .....	39
Mandates .....	39
Mandate Download .....	39
Request Sample .....	39
Response Sample.....	40
Products .....	40
Product Fields .....	41
Get Product List .....	41
Request Sample .....	41
Response Sample.....	41
Get Product Details.....	42
Request Sample .....	42
Response Sample.....	42
Insert Product.....	42
Request Sample .....	42
Response Fields .....	43
Response Samples .....	43
Update Product .....	44
Request Sample .....	44
Response Fields .....	44
Response Sample.....	44
Transaction Operations .....	45
Transactions .....	45
Transaction Fields.....	45
Get Transaction List By CollectionID.....	45
Get Transaction List By Transaction Date .....	47
Get Transaction List By Transaction Date Range.....	48
Unpays .....	49
Unpaid Transaction Fields.....	49

Get Unpays By Collection.....	50
Get Unpays By Transaction Date.....	51
Get Unpays By Transaction Date Range .....	51
AVS Operations .....	52
Request Avs Record .....	52
Request Fields.....	53
Response Fields .....	54
Client Avs Callback Implementation Guide.....	55
Example Project .....	55
CDV (Check Digit Verification).....	56
Request Fields.....	56
Response Fields .....	56
Request Sample .....	57
Table Types and Definitions.....	57
Client Status Types.....	57
Client Types .....	57
Address Types.....	57
Collection Status Types .....	58
Collection Frequency Types .....	58
Collection Method Types .....	58
Account Types .....	58
NAEDO Tracking Codes .....	59
Generation Types.....	59
Transaction Types .....	59
Unpaid Reason Types.....	60

## INTRODUCTION

The Softy Comp API (SAPI) is one of the integrated channel solutions. It uses a secure, automated, two-way data transfer service that renders products and services to entities outside Softy Comp through a digital channel.

The channel will allow clients to interface their in-house systems directly and securely with that of Softy Comp.

The SAPI channel will offer the following services:

- Debit order collections
- EFT
- NAEDO

This document will focus on the integrations and method implementation to the different services available. It also covers the message and request structures for different services and provides sample codes. It will provide the links to the service contract definitions that users must adopt in order to cater for the services provided by SAPI.

SAPI is a Restful JSON services that run over HTTPS.

# API METHODS

## LATEST API DOCUMENTATION

To get the latest api document please paste the following in your browser to download it:

<https://scoretest.softycomp.co.za/SoftyCompBureauAPI/api/sapidocuments/getlatestsapidocument>

## API FIELDS AND RULES

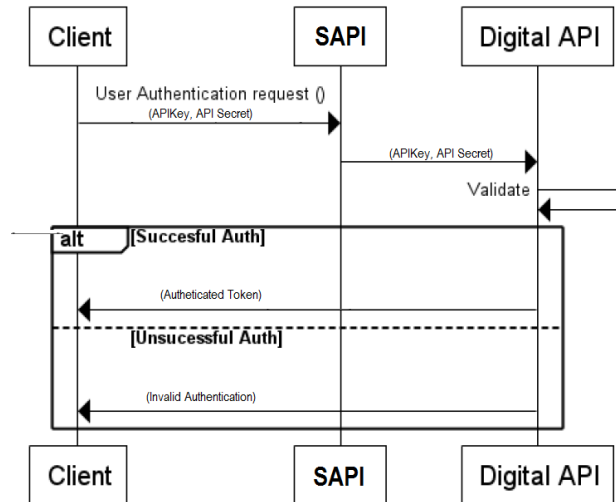
In this section the industry level data fields and rules around it will be provided for reference purposes. The following abbreviations are used.

### FIELDS

Field Type	Required Types
S = String	M = Mandatory
N = Number	O = Optional
B = Boolean	C = Conditional
I = Integer	
DTO = Data Transfer Object (Array)	
DT = Date	

## AUTENTICATION OPERATIONS

### AUTHENTICATION USER SEQUENCE DIAGRAM



### AUTHENTICATION USER REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
APIKey	S	M	50	The apiKey provided during on boarding.
APISecret	S	M	50	The apiSecret provided during on boarding.

### SAMPLE CODE FOR AUTHETICATION REQUEST

URL	<b>Provided URL</b> /api/auth/generatetoken
Method	POST
Header	Content-Type: application/json
Request	<pre> {   "apiKey": "Provided_APIKey",   "apiSecret": "Provided_APISecret" } </pre>





Initials	S	3	The initials of the clients if the client is not a company.
Surname	S	120	The client surname or the company name.
IDNumber	S	30	If the client is not a company and the ID number is mandatory then the fields needs to be filled in. If no the field can be left empty.
Registration Number		30	If the client is a company and the registration number is mandatory then the fields needs to be filled in. If no the field can be left empty.
ClientStatusTypeID	N	-	Indicates the status of the client.
HomeTelephoneNumber	S	10	The home telephone number.
WorkTelephoneNumber	S	10	The work place telephone number.
CellphoneNumber	S	10	If the cell phone number is not mandatory this field can be left empty. If the field is filled in the format is as follows: 0722733216
EmailAddress	S	90	The clients email address.
SendSmsDoNotifications	B	-	Allow SMS debit order notifications for this client.
SendSmsUnpaidNotifications	B	-	Allow SMS unpaid notifications for this client.
AddressTypeID	N	-	The Address type of the client. Default to 1(street).
Address	S	100	Address details of the client.
AddressSuburb	S	250	The suburb of the address.
AddressTown	S	250	The town of the address.
Province	S	100	The province of the address.
PostalCode	S	6	The postal code of the address.
InceptionDate	DT	-	This is an optional field for your own use, and can contain the date the contract started, or client added.
Vatno	S	50	The vat number.
IsSouthAfricanCitizen	B	-	True/False if the client is a South African Citizen.
AllowInvoicing	B	-	- Allows Invoices to be sent to the client. This field will default to false if the applicable company settings are false. - If not specified it will have a default value of false
AdditionalInfo	S	MAX	

#### CLIENT LIST FILTERING REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
ClientFilter	DTO	M	-	Specify to filter the client list
EmailAddress	S	O	90	The clients email address.
CellphoneNumber	S	O	10	Client Cellphone Number. If the field is filled in the format is as follows: 0722733216
ContractCode	S	O	8	Client Contract Code
ClientStatusTypeID	N	O	-	Specify to filter the client list

TIP: The Filter request body is required, but if you do not want to apply any filtering then simply leave the fields as empty strings.

---

 REQUEST SAMPLE

URL	<i>Example: <b>Provided URL</b>/api/clients/listclients</i>
Method	GET or POST
Headers	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "emailAddress": "",   "cellphoneNumber": "",   "contractcode": "",   "clientStatusTypeID": "", }</pre>

---

 RESPONSE SAMPLE

Sample
<pre>[   {     "clientId": 1237305,     "clientTypeId": 2,     "contractCode": "API01",     "initials": "MK",     "surname": "Rosewall",     "idnumber": "9006210009088",     "registrationNumber": "",     "clientStatusTypeID": 1,     "homeTelephoneNumber": "",     "workTelephoneNumber": "",     "cellphoneNumber": "0748891529",     "emailAddress": "anel@softyycomp.co.za",     "sendSmsDonotifications": false,     "sendSmsUnpaysNotifications": false,     "isSouthAfricanCitizen": true,     "addressTypeId": 1,     "address": "1 Test Streets",     "addressSuburb": "Eldoraigne",     "addressTown": "Centurion",     "postalCode": "0157",     "inceptionDate": "2019-07-02T00:00:00",     "vatno": "",     "allowInvoicing": false,     "province": "Gauteng",     "additionalInfo": "Testing Client used for API"   },   {     "clientId": 1237307,     "clientTypeId": 2,     "contractCode": "API02",</pre>

```

    "initials": "MK",
    "surname": "Rosewall",
    "idnumber": "9006210009088",
    "registrationNumber": "",
    "clientStatusTypeId": 1,
    "homeTelephoneNumber": "",
    "workTelephoneNumber": "",
    "cellphoneNumber": "0748891529",
    "emailAddress": "anel@softyycomp.co.za",
    "sendSmsDonotifications": false,
    "sendSmsUnpaysNotifications": false,
    "isSouthAfricanCitizen": true,
    "addressTypeId": 1,
    "address": "1 Test Street",
    "addressSuburb": "Eldoraigne",
    "addressTown": "Centurion",
    "postalCode": "0157",
    "inceptionDate": "2019-07-02T00:00:00",
    "vatno": "",
    "allowInvoicing": false,
    "province": "Gauteng",
    "additionalInfo": "Testing Client 2 used for API"
  }
]

```

## GET CLIENT DETAILS

This method allows the user to get a specific client on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/clients/getclientdetails/{clientID} <i>Example: <b>Provided URL</b> /api/clients/getclientdetails/1237305</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

### RESPONSE SAMPLE

Sample
<pre> {   "clientId": 1237305,   "clientId": 1237305,   "clientTypeId": 2,   "contractCode": "API01",   "initials": "MK",   "surname": "Rosewall", </pre>

```

{idnumber": "9006210009088",
"registrationNumber": "",
"clientStatusTypeld": 1,
"homeTelephoneNumber": "",
"workTelephoneNumber": "",
"cellphoneNumber": "0748891529",
"emailAddress": "anel@softyycomp.co.za",
"sendSmsDonotifications": false,
"sendSmsUnpaysNotifications": false,
"isSouthAfricanCitizen": true,
"addressTypeld": 1,
"address": "1 Test Street",
"addressSuburb": "Eldoraigne",
"addressTown": "Centurion",
"postalCode": "0157",
"inceptionDate": "2019-07-02T00:00:00",
"vatno": "",
"allowInvoicing": false,
"province": "Gauteng",
"additionalInfo": "Testing Client used for API"
}

```

**GET CLIENT HISTORY DETAILS**

This method allows the user to get a specific client on their profile.

REQUEST SAMPLE

URL	<b>Provided URL</b> /api/clients/getclienthistorydetails/{clientID} <i>Example: <b>Provided URL</b> /api/clients/getclienthistorydetails/1237305</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

RESPONSE SAMPLE

Sample
<pre> { "clientHistoryID": 260735346, "clientID": 1, "auditDate": "2018-08-10T14:57:55.06", "userID": 1, "columnName": "ContractCode", "previousValue": "0000001", "newValue": "S_S 1" } </pre>

## INSERT CLIENT

This method allows the user to add a new client on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/clients/createclient
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "clientId": 0,   "clientTypeId": 1,   "contractCode": "000033",   "initials": "",   "surname": "Company B",   "idnumber": "",   "registrationNumber": "1/123/1234",   "clientStatusTypeId": 1,   "homeTelephoneNumber": "",   "workTelephoneNumber": "0123456789",   "cellphoneNumber": "0123456789",   "emailAddress": "test@test.co.za",   "sendSmsDonotifications": false,   "sendSmsUnpaysNotifications": false,   "isSouthAfricanCitizen": true,   "addressTypeId": 2,   "address": "123",   "addressSuburb": "ABC East",   "addressTown": "Moot",   "postalCode": "0039",   "inceptionDate": "2017-05-10T00:00:00",   "vatno": null,   "allowInvoicing": false,   "province": "Not Selected",   "additionalInfo": null }</pre>

### RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Client ID. If it was a new client being created it will return the new Client ID
Success	B	-	Indicates if the Client was saved or not
Messages	DTO	-	Will show a list of errors if Client did not save successfully. Will be empty if client saved successfully

---

RESPONSE SAMPLES

---

SUCCESSFUL RESPONSE

Sample
<pre>{   "value": 1237307,   "success": true,   "messages": [] }</pre>

---

ERROR RESPONSE

Sample
<pre>{   "value": 0,   "success": false,   "messages": [     "The Client Type ID '0' does not exist."   ] }</pre>

UPDATE CLIENT

This method allows the user to update a specific client on their profile.

---

REQUEST SAMPLE

URL	<b>Provided URL</b> /api/clients/updateclient
Method	POST
Header	Content-Type: application/json Authorization: <b>Generated Bearer Token</b>
Request	<pre>{   "clientId": 1237305,   "clientId": 1237305,   "clientTypeId": 2,   "contractCode": "API01",   "initials": "MK",   "surname": "Rosewall",   "idnumber": "9006210009088",   "registrationNumber": "",   "clientStatusTypeId": 1,   "homeTelephoneNumber": "",   "workTelephoneNumber": "",   "cellphoneNumber": "0748891529",   "emailAddress": "anel@softycomp.co.za",   "sendSmsDonotifications": false,   "sendSmsUnpaysNotifications": false,   "isSouthAfricanCitizen": true,   "addressTypeId": 1,   "address": "1 Test Streets", }</pre>

```

"addressSuburb": "Eldoraigne",
"addressTown": "Centurion",
"postalCode": "0157",
"inceptionDate": "2019-07-02T00:00:00",
"vatno": "",
"allowInvoicing": false,
"province": "Gauteng",
"additionalInfo": "Testing Client used for API"
}

```

RESPONSE FIELDS

Field Name	Type	Length	Description
value	N	-	This will contain the Client ID. If it was a new client being created it will return the new Client ID
success	B	-	Indicates if the Client was saved or not
messages	DTO	-	Will show a list of errors if Client did not save successfully. Will be empty if client saved successfully.

RESPONSE SAMPLE

SUCCESSFUL RESPONSE

Sample
<pre> {   "value": 1237305,   "success": true,   "messages": [] } </pre>

ERROR RESPONSE

Sample
<pre> {   "value": 1237305,   "success": false,   "messages": [     "The Client Type ID '0' does not exist."   ] } </pre>



## COLLECTION OPERATIONS

### GET COLLECTION LIST

This method allows the user to list all the collections for a specific client on their profile.

#### COLLECTION FIELDS

Field Name	Type	Length	Description
Collection	DTO	-	Collection Data Transfer Object
CollectionID	N	-	The unique system ID.
CollectonMethodTypeID	N	-	The collection method.
CollectionFrequencyTypeID	N	-	The frequency of the collection.
CommencementDate	DT	-	The date on which the debit order will start.
TransactionDate	DT	-	The transaction date of the debit order if the frequency is a once off.
TransactionDate2	DT	-	The 2 <sup>nd</sup> transaction date of the debit order if the frequency is 2x once off.
TransactionDate3	DT	-	The 3 <sup>rd</sup> transaction date of the debit order if the frequency is 4x once off.
TransactionDate4	DT	-	The 4 <sup>th</sup> transaction date of the debit order if the frequency is 4x once off.
ExpiryDate	DT	-	The expiry date of the debit order. If left empty the debit order will not expire.
DebitDay	N	-	The debit day of the month the debit order will go off if the relevant frequency is selected. (1-31)
DebitMonth	N	-	The debit month of the year the debit order will go off if the relevant frequency is selected. (112)
Amount	N	-	The amount that will be collected.
ToCollectionAmountTotal	N	-	The to collect amount total is the total that will be collected before the debit order stops.
NaedoTrackingCodeID	N	-	The tracking code for the amount of days the Naedo debit order will track. Default to 12 of Eft.
CollectionDescription	S	50	User custom description.
AccountTypeID	N	-	The account type for the banking details of the collection.
BranchCode	N	-	The bank branch code.
AccountNumber	S	20	The collection account number.
AccountName	S	70	The account name of the collection.
ProductID	N	-	The Id of the product of this Client. 1 is the default product. Value must be 1 or greater.
MandateDTO	DTO	-	Mandate Data Transfer Object

MandateID	N	-	The Unique ID of the Mandate Document
FileName	S	Max	Name of the mandate document
DocumentGUID	S	36	Globally Unique Identifier of the mandate document
UploadedDate	DT	-	The Date when the document was uploaded.
FileData	byte[]	-	Actual Content data of the file.

---

**REQUEST SAMPLE**

URL	<b>Provided URL</b> /api/collections/listcollections/{clientID} <i>Example: <b>Provided URL</b>/api/collections/listcollections/1237305</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

---

**RESPONSE SAMPLE**

Sample
<pre>[   {     "collectionId": 1627206,     "clientId": 1237305,     "collectionMethodTypeId": 4,     "collectionFrequencyTypeId": 1,     "collectionStatusTypeId": 1,     "commencementDate": "2019-07-09T00:00:00",     "transactionDate": "2019-07-12T00:00:00",     "expiryDate": null,      "debitDay": null,     "debitMonth": null,     "amount": 55,     "toCollectionAmountTotal": null,     "naedoTrackingCodeId": 12,     "collectionDescription": "Test",     "accountTypeId": 1,     "branchCode": 250655,     "accountNumber": "12345678901",     "accountName": "MK Rosewalll",     "transactionDate2": null,     "transactionDate3": null,   } ]</pre>

```

    "transactionDate4": null,
    "productId": 1,
    "mandate": {
      "fileName": "Test.PNG",
      "mandateId": 573446,
      "documentGUID": "aeaa90e6-9d7c-4cc0-ba1e-62ac10f8fa43",
      "uploadedDate": "2019-07-09T08:30:06.817"
    }
  },
  {
    "collectionId": 1627207,
    "clientId": 1237305,
    "collectionMethodTypeId": 4,
    "collectionFrequencyTypeId": 1,
    "collectionStatusTypeId": 1,
    "commencementDate": "2019-07-02T00:00:00",
    "transactionDate": "2019-07-06T00:00:00",
    "expiryDate": null,
    "debitDay": null,
    "debitMonth": null,
    "amount": 40,
    "toCollectionAmountTotal": null,
    "naedoTrackingCodeId": 12,
    "collectionDescription": "Test",
    "accountTypeId": 1,
    "branchCode": 250655,
    "accountNumber": "12345678901",
    "accountName": "MK Rosewall",
    "transactionDate2": null,
    "transactionDate3": null,
    "transactionDate4": null,
    "productId": 1,
    "mandate": {
      "fileName": null,
      "mandateId": 0,
      "documentGUID": null,
      "uploadedDate": null
    }
  }
]

```

### GET COLLECTION FILTER LIST

This method allows the user to list all the collections for a specific client on their profile.

#### COLLECTION FIELDS

Field Name	Type	Length	Description
Collection	DTO	-	Collection Data Transfer Object
CollectionID	N	-	The unique system ID.
CollectionMethodTypeID	N	-	The collection method.
CollectionFrequencyTypeID	N	-	The frequency of the collection.

CommencementDate	DT	-	The date on which the debit order will start.
TransactionDate	DT	-	The transaction date of the debit order if the frequency is a once off.
TransactionDate2	DT	-	The 2 <sup>nd</sup> transaction date of the debit order if the frequency is 2x once off.
TransactionDate3	DT	-	The 3 <sup>rd</sup> transaction date of the debit order if the frequency is 4x once off.
TransactionDate4	DT	-	The 4 <sup>th</sup> transaction date of the debit order if the frequency is 4x once off.
ExpiryDate	DT	-	The expiry date of the debit order. If left empty the debit order will not expire.
DebitDay	N	-	The debit day of the month the debit order will go off if the relevant frequency is selected. (1-31)
DebitMonth	N	-	The debit month of the year the debit order will go off if the relevant frequency is selected. (1-12)
Amount	N	-	The amount that will be collected.
ToCollectionAmountTotal	N	-	The to collect amount total is the total that will be collected before the debit order stops.
NaedoTrackingCodeID	N	-	The tracking code for the amount of days the Naedo debit order will track. Default to 12 of Eft.
CollectionDescription	S	50	User custom description.
AccountTypeID	N	-	The account type for the banking details of the collection.
BranchCode	N	-	The bank branch code.
AccountNumber	S	20	The collection account number.
AccountName	S	70	The account name of the collection.
ProductID	N	-	The Id of the product of this Client. 1 is the default product. Value must be 1 or greater.
MandateDTO	DTO	-	Mandate Data Transfer Object
MandateID	N	-	The Unique ID of the Mandate Document
FileName	S	Max	Name of the mandate document
DocumentGUID	S	36	Globally Unique Identifier of the mandate document
UploadedDate	DT	-	The Date when the document was uploaded.
FileData	byte[]	-	Actual Content data of the file.

## REQUEST SAMPLE

URL	<b>Provided URL</b> /api/collections/listcollectionsfilter /{clientID} <i>Example: <b>Provided URL</b>/api/collections/listcollections/1237305</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	{ "CollectionStatusTypeId": "", }

## RESPONSE SAMPLE

Sample
<pre>[   {     "collectionId": 1627206,     "clientId": 1237305,     "collectionMethodTypeId": 4,     "collectionFrequencyTypeId": 1,     "collectionStatusTypeId": 2,     "commencementDate": "2019-07-09T00:00:00",     "transactionDate": "2019-07-12T00:00:00",     "expiryDate": null,     "debitDay": null,     "debitMonth": null,     "amount": 55,     "toCollectionAmountTotal": null,     "naedoTrackingCodeId": 12,     "collectionDescription": "Test",     "accountTypeId": 1,     "branchCode": 250655,     "accountNumber": "12345678901",     "accountName": "MK Jackson",     "transactionDate2": null,     "transactionDate3": null,     "transactionDate4": null,     "productId": 1,     "mandate": {       "fileName": "Test.PNG",       "mandateId": 573446,       "documentGUID": "aeaa90e6-9d7c-4cc0-ba1e-62ac10f8fa43",       "uploadedDate": "2019-07-09T08:30:06.817"     }   } ]</pre>

## GET ALL COLLECTION LIST

This method allows the user to list all the collections for a specific client on their profile.

---

**COLLECTION FIELDS**

Field Name	Type	Length	Description
Collection	DTO	-	Collection Data Transfer Object
CollectionID	N	-	The unique system ID.
CollectionMethodTypeID	N	-	The collection method.
CollectionFrequencyTypeID	N	-	The frequency of the collection.
CommencementDate	DT	-	The date on which the debit order will start.
TransactionDate	DT	-	The transaction date of the debit order if the frequency is a once off.
TransactionDate2	DT	-	The 2 <sup>nd</sup> transaction date of the debit order if the frequency is 2x once off.
TransactionDate3	DT	-	The 3 <sup>rd</sup> transaction date of the debit order if the frequency is 4x once off.
TransactionDate4	DT	-	The 4 <sup>th</sup> transaction date of the debit order if the frequency is 4x once off.
ExpiryDate	DT	-	The expiry date of the debit order. If left empty the debit order will not expire.
DebitDay	N	-	The debit day of the month the debit order will go off if the relevant frequency is selected. (1-31)
DebitMonth	N	-	The debit month of the year the debit order will go off if the relevant frequency is selected. (1-12)
Amount	N	-	The amount that will be collected.
ToCollectionAmountTotal	N	-	The to collect amount total is the total that will be collected before the debit order stops.
NaedoTrackingCodeID	N	-	The tracking code for the amount of days the Naedo debit order will track. Default to 12 of Eft.
CollectionDescription	S	50	User custom description.
AccountTypeID	N	-	The account type for the banking details of the collection.
BranchCode	N	-	The bank branch code.
AccountNumber	S	20	The collection account number.
AccountName	S	70	The account name of the collection.
ProductID	N	-	The Id of the product of this Client. 1 is the default product. Value must be 1 or greater.
MandateDTO	DTO	-	Mandate Data Transfer Object
MandateID	N	-	The Unique ID of the Mandate Document
FileName	S	Max	Name of the mandate document
DocumentGUID	S	36	Globally Unique Identifier of the mandate document
UploadedDate	DT	-	The Date when the document was uploaded.
FileData	byte[]	-	Actual Content data of the file.

---

 REQUEST SAMPLE

URL	<b>Provided URL</b> /api/collections/listallcollections <i>Example: <b>Provided URL</b>/api/collections/listallcollections</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

---

 RESPONSE SAMPLE

Sample
<pre>[   {     "collectionId": 1627206,     "clientId": 1237305,     "collectionMethodTypeId": 4,     "collectionFrequencyTypeId": 1,     "collectionStatusTypeId": 1,     "commencementDate": "2019-07-09T00:00:00",     "transactionDate": "2019-07-12T00:00:00",     "expiryDate": null,     "debitDay": null,     "debitMonth": null,     "amount": 55,     "toCollectionAmountTotal": null,     "naedoTrackingCodeId": 12,     "collectionDescription": "Test",     "accountTypeId": 1,     "branchCode": 250655,     "accountNumber": "12345678901",     "accountName": "MK Rosewall",     "transactionDate2": null,     "transactionDate3": null,     "transactionDate4": null,     "productId": 1,   },   {     "collectionId": 1627207,     "clientId": 1237305,     "collectionMethodTypeId": 4,     "collectionFrequencyTypeId": 1,     "collectionStatusTypeId": 1,     "commencementDate": "2019-07-02T00:00:00",     "transactionDate": "2019-07-06T00:00:00",     "expiryDate": null,     "debitDay": null,</pre>

```

    "debitMonth": null,
    "amount": 40,
    "toCollectionAmountTotal": null,
    "naedoTrackingCodeId": 12,
    "collectionDescription": "Test",
    "accountTypeId": 1,
    "branchCode": 250655,
    "accountNumber": "12345678901",
    "accountName": "MK Rosewall",
    "transactionDate2": null,
    "transactionDate3": null,
    "transactionDate4": null,
    "productId": 1,
  }
]

```

### GET COLLECTION DETAILS

This method allows the user to get the details for on their profile.

#### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/collections/getcollectiondetails /{collectionID} Example: <b>Provided URL</b> /api/collections/getcollectiondetails/1627206
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

#### RESPONSE SAMPLE

Sample
<pre> {   "collectionId": 1627206,   "clientId": 1237305,   "collectionMethodTypeId": 4,   "collectionFrequencyTypeId": 1,   "collectionStatusTypeId": 1,   "commencementDate": "2019-07-09T00:00:00",   "transactionDate": "2019-07-12T00:00:00",   "expiryDate": null,   "debitDay": null,   "debitMonth": null,   "amount": 55,   "toCollectionAmountTotal": null,   "naedoTrackingCodeId": 12,   "collectionDescription": "Test",   "accountTypeId": 1, </pre>



```

"branchCode": 250655,
"accountNumber": "12345678901",
"accountName": "MK Rosewalll",
"transactionDate2": null,
"transactionDate3": null,
"transactionDate4": null,
"productId": 1,
"mandate": {
  "fileName": "Test.PNG",
  "mandateId": 573446,
  "documentGUID": "aeaa90e6-9d7c-4cc0-ba1e-62ac10f8fa43",
  "uploadedDate": "2019-07-09T08:30:06.817"
}
}

```

## GET COLLECTION HISTORY DETAILS

This method allows the user to get the history details for on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/collections/ getcollectionhistorydetails/{collectionID} Example: <b>Provided URL</b> /api/collections/ getcollectionhistorydetails /1627206
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

### RESPONSE SAMPLE

Sample
<pre> [   {     "collectionHistoryID": 736310775,     "collectionID": 910,     "auditDate": "2018-10-31T08:11:50.69",     "userID": 48,     "columnName": "CommencementDate",     "previousValue": "2016-08-05",     "newValue": "2018-11-03"   } ] </pre>

## CREATE COLLECTION

This method adds a new collection to an existing client. Please note that collection rules apply.

### REQUEST SAMPLE

URL	<i>Example: <b>Provided URL</b>/api/collections/createcollection</i>
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "clientId": 1237305,   "collectionMethodTypeId": 4,   "collectionFrequencyTypeId": 1,   "collectionStatusTypeId": 1,   "commencementDate": "2019-07-09 00:00:00",   "transactionDate": "2019-07-12 00:00:00",   "expiryDate": null,   "debitDay": null,   "debitMonth": null,   "amount": 52,   "toCollectionAmountTotal": null,   "naedoTrackingCodeId": 12,   "collectionDescription": "Test",   "accountTypeId": 1,   "branchCode": 250655,   "accountNumber": "12345678901",   "accountName": "MK Rosewall",   "transactionDate2": null,   "transactionDate3": null,   "transactionDate4": null,   "productId": 1,   "mandate": {     "Filename": "Test.PNG",     "FileData": "0x89504E470D0A1A0A0000000D494844520000011A000000DF 080600000029E11C09000000017352474200AECE1CE90000000467414D4100 00B18 F0BFC61050000000970485973000012740000127401DE661F78000 0058849444154785EEDDD8D6DD44A1840D1B4400DB4400F94400DB44007E 9800EA8800A688006E8801E160D8F41665F40F1666F98C5E7489650B4760CA 3EFCAFBE3E5EE0410131A202734404E68809CD00039A101724203E48406C80 90D90131A202734404E68809CD00039A101724203E48406C8090D90131A20 2734404E68809CD00039A101724203E48406C8090D90131A202734404E68...</pre>

### RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Collection ID. If it was a new Collection being created it will return the new Collection ID

Success	B	-	Indicates if the Client was saved or not.
Messages	DTO	-	Will show a list of errors if Collection did not save successfully. Will be empty if Collection saved successfully

---

## RESPONSE SAMPLES

### SUCCESSFUL RESPONSE

Sample
<pre>{   "value": 1627208,   "success": true,   "messages": [] }</pre>

### ERROR RESPONSE

Sample
<pre>{   "value": 0,   "success": false,   "messages": [     "The Collection Method Type ID '0' does not exist."   ] }</pre>

## UPDATE COLLECTION

This method updates a collection on an existing client. Please note that collection rules apply.

### REQUEST SAMPLE

URL	<i>Example: <b>Provided URL</b>/api/collections/updatecollection</i>
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "collectionId": 1627206,   "clientId": 1237305,   "collectionMethodTypeId": 4,   "collectionFrequencyTypeId": 1,   "collectionStatusTypeId": 1,   "commencementDate": "2019-07-09 00:00:00",   "transactionDate": "2019-07-12 00:00:00",   "expiryDate": null,   "debitDay": null,   "debitMonth": null,   "amount": 55,   "toCollectionAmountTotal": null, }</pre>

```

"naedoTrackingCodeId": 12,
"collectionDescription": "Test",
"accountTypeId": 1,
"branchCode": 250655,
"accountNumber": "12345678901",
"accountName": "MK Rosewalll",
"transactionDate2": null,
"transactionDate3": null,
"transactionDate4": null,
"productId": 1,
"mandate":{
  "Filename": "Test.PNG",
  "FileData":
  "0x89504E470D0A1A0A0000000D49484452000011A000000DF0806000002
  9E11C09000000017352474200AECE1CE90000000467414D410000B18F0BFC6
  1050000000970485973000012740000127401DE661F7800000588494441547
  85EEDDD..."
}
}

```

RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Collection ID. If it was a new Collection being created it will return the new Collection ID
Success	B	-	Indicates if the Client was saved or not.
Messages	DTO	-	Will show a list of errors if Collection did not save successfully. Will be empty if Collection saved successfully

RESPONSE SAMPLES

SUCCESSFUL RESPONSE

Sample
<pre> {   "value": 1627206,   "success": true,   "messages": [] } </pre>

ERROR RESPONSE

Sample

```
{
  "value": 1627206,
  "success": false,
  "messages": [
    "The provided transaction date cannot be smaller than 2019-07-11!"
  ]
}
```

## DEBICHECK COLLECTIONS

### DEBICHECK COLLECTION FIELDS

Field Name	Type	Required	Length	Description
Mandate	DTO	-	-	Mandate Data Transfer Object
AuthenticationInstrument	S	M	4	<b>Authentication Instrument:</b> <ul style="list-style-type: none"> <li>- This field is used to indicate the debtor authorisation method code.</li> <li>- Expected values: 0227, 0229 or 0230</li> <li>- For TT1 delayed use code 0227</li> <li>- For TT1 real-time use codes 0229 and 0230</li> <li>- When using code 0229, the debtor cell number in the message must match the registered cell number at the paying bank. If cell number provided in the message does not match, the request will be rejected.</li> <li>- When using code 0230, if the cell number in the message does not match the registered cell number at the paying bank, authorisation will be sent to payer using the registered cell number.</li> </ul>
TransmissionNumber	I	M	1	<ul style="list-style-type: none"> <li>- This field is used to indicate the number of times the transaction has been sent</li> <li>- The client is allowed to resubmit a failed transaction up to 4 times using the same ClientRequestReference and ClientContractReference.</li> <li>- If after 4 times the mandate initiation is still not successful, the client has to change the ClientRequestReference number</li> <li>- Must contain the number of times a message with this message identifier has been transmitted (in case of re-tries):</li> <li>- 1 = original message</li> <li>- 2 = resent message</li> <li>- 3 = resent message</li> <li>- 4 = resent message</li> </ul>
CallbackUrl	S	M	1000	<ul style="list-style-type: none"> <li>- The call-back URL to where the results for the record will be pushed to when it comes through.</li> </ul>
FirstCollectionDate	DT	O	-	<ul style="list-style-type: none"> <li>- The date on which the first collection (initial amount) will occur.</li> </ul>

				<ul style="list-style-type: none"> <li>- If the First Collection Date is populated, then the initial amount must also be populated.</li> <li>- The "First Collection Date" (if populated) must be at least 2 days into the future including current day and must be a valid date.</li> <li>- Format: YYYY-MM-DDT00:00:00.000Z.</li> </ul>
AdjustmentCategoryTypeID	N	M	2	<ul style="list-style-type: none"> <li>- Refers to the ability to adjust the Instalment Amount and the Maximum Collection Amount.</li> <li>- Expected values:</li> <li>- Never = 10</li> <li>- Quarterly = 20</li> <li>- Biannually = 30</li> <li>- Annually = 40</li> <li>- Repo = 50</li> <li>- If the Adjustment Category is "Quarterly" or "Biannually" or "Annually", then either use "UseAmountAdjustment" and "AdjustmentAmount" or "UseRateAdjustment" and "AdjustmentRate".</li> <li>- If the Adjustment Category is "Never" or "Repo" then "UseAmountAdjustment", "AdjustmentAmount", "UseRateAdjustment" and "AdjustmentRate" are not required.</li> </ul>
UseAdjustmentAmount	B	M	1	<ul style="list-style-type: none"> <li>- An indicator to determine if the Adjustment Amount will be used.</li> <li>- Expected values for Boolean:</li> <li>- 0 (False)</li> <li>- 1 (True)</li> <li>- If the Adjustment Category is "Quarterly" or "Biannually" or "Annually", then either use "UseAmountAdjustment" and "AdjustmentAmount" or "UseRateAdjustment" and "AdjustmentRate"</li> </ul>
AdjustmentAmount	N	M	-	<ul style="list-style-type: none"> <li>- Amount by which the Instalment Amount and the Maximum Collection Amount can be adjusted based on Adjustment Category. This value can be negative.</li> <li>- If the Adjustment Category is "Quarterly" or "Biannually" or "Annually" then either use "UseAmountAdjustment" and "AdjustmentAmount" or "UseRateAdjustment" and "AdjustmentRate".</li> <li>- Adjustment Amount to increase per Adjustment Category.</li> </ul>
UseAdjustmentRate	B	M	1	<ul style="list-style-type: none"> <li>- Expected values for Boolean:</li> <li>- 0 (False)</li> <li>- 1 (True)</li> <li>- If the Adjustment Category is "Quarterly" or "Biannually" or "Annually" then either use UseAmountAdjustment and AdjustmentAmount or UseRateAdjustment and AdjustmentRate.</li> </ul>
AdjustmentRate	N	M	8	<ul style="list-style-type: none"> <li>- Rate by which the Instalment Amount and the Maximum Collection Amount can be adjusted based on Adjustment Category.</li> <li>- This value can be negative.</li> <li>- Adjustment Rate to increase or decrease per Adjustment Category.</li> </ul>

				<ul style="list-style-type: none"> <li>- If the Adjustment Category is “Quarterly” or “Biannually” or “Annually” then either use “UseAmountAdjustment” and “AdjustmentAmount” or “UseRateAdjustment” and “AdjustmentRate”.</li> <li>- If adjustment category is equal to “Never” or “Repo”, neither the Adjustment Rate nor the Adjustment Amount is required.</li> </ul>
CollectionDay	I	M	2	<ul style="list-style-type: none"> <li>- Contains a number for the day of the week or a day of the month as per the frequency selected in “Frequency Field”.</li> <li>- Collection Day should be in the range of 1 – 99.</li> <li>- Refer to Mandate frequency codes table.</li> </ul>
InitiationDate	DT	M	-	<ul style="list-style-type: none"> <li>- Mandate initiation date.</li> <li>- Must be the current date</li> <li>- The date must be a valid date.</li> <li>- It may not be blank or have spaces between characters.</li> <li>- Format: YYYY-MM-DDT00:00:00.000Z</li> </ul>
DateAdjustmentAllowed	B	M	1	<ul style="list-style-type: none"> <li>- Used to indicate that the Collection Day could change.</li> <li>- Expected values for Boolean: <ul style="list-style-type: none"> <li>- 0 (False)</li> <li>- 1 (True)</li> </ul> </li> <li>- True “1” - Allow all transactions for processing irrespective of the Collection Date.</li> <li>- False “0” - Allow transaction if Action Date = Collection Day. Upfront rejection if Action Date ≠ Collection Day.</li> </ul>
MandateReleaseDate	DT	O	10	<ul style="list-style-type: none"> <li>- Indicate the date on which the mandate can be released, e.g. the mandate can be captured today to be released in the future for debtor authentication.</li> <li>- Mandate Release Date must be a future date.</li> <li>- Format: YYYY-MM-DDT00:00:00.000Z.</li> </ul>
TrackingIndicator	B	O	1	<ul style="list-style-type: none"> <li>- Specify if tracking may be used for collections.</li> <li>- Expected values for Boolean: <ul style="list-style-type: none"> <li>- 0 (False)</li> <li>- 1 (True)</li> </ul> </li> <li>- If Tracking Indicator is “true”, then tracking is enabled – this means that the debtor has given ABSA the authority to track the account in the event of a collection failure.</li> <li>- This also means that for up to 10 days following the collection failure, ABSA can monitor the debtor’s account, and as soon as money comes into the account, the collection can be resubmitted.</li> <li>- If Tracking Indicator is “false” – then debtor has refused to allow the monitoring of his account.</li> <li>- If not provided, will default to false.</li> </ul>
InstalmentOccurrenceTypeID	I	M	1	<ul style="list-style-type: none"> <li>- Is the sequence type for instalment occurrence.</li> <li>- Expected values: <ul style="list-style-type: none"> <li>- Once-off = 1</li> </ul> </li> </ul>

				<ul style="list-style-type: none"> <li>- Recurring = 2</li> </ul>
InstalmentAmount	N	M	15	<ul style="list-style-type: none"> <li>- This is the regular collection amount.</li> <li>- Instalment Amount may not be greater than "Maximum Collection Amount".</li> </ul>
MaximumCollectionAmount	N	M	15	<ul style="list-style-type: none"> <li>- The maximum collection amount that the user/creditor is allowed to collect per debit payment.</li> <li>- The Maximum Collection Amount cannot be greater than 1,5 times the Instalment Amount.</li> <li>- Example: If R100.00 is the Instalment Amount, the Maximum Collection Amount may not exceed R150.00.</li> <li>- The Maximum Collection Amount must be equal to or greater than the Instalment Amount.</li> <li>- The field may not be blank or have spaces between characters and must be greater than "0".</li> </ul>
UseInitialAmount	B	M	1	<ul style="list-style-type: none"> <li>- Indicator to determine if there is an Initial Amount provided that should be included in the mandate:</li> <li>- Expected values for Boolean:</li> <li>- 0 (False)</li> <li>- 1 (True)</li> </ul>
InitialAmount	N	M	15	<ul style="list-style-type: none"> <li>- The Initial Amount to be collected.</li> </ul>
FrequenceTypeID	I	M	3	<ul style="list-style-type: none"> <li>- Used to indicate the regularity of the collection, namely: weekly, fortnightly, monthly, quarterly, annually, biannually or monthly by rule (ad hoc).</li> <li>- Expected values:</li> <li>- Weekly = 2</li> <li>- Fortnightly = 4</li> <li>- Monthly = 8</li> <li>- Quarterly = 16</li> <li>- Biannually = 32</li> <li>- Yearly = 64</li> <li>- Ad Hoc = 128</li> <li>- The Collection Day must align with the frequency required in the table of mandate frequency.</li> </ul>
EntryClassCodeTypeID	I	M	4	<ul style="list-style-type: none"> <li>- Type of debit as determined by the creditor.</li> <li>- Please see list of Entry Class Code Types for valid ID's.</li> </ul>
DebtorAccountName	S	M	35	<ul style="list-style-type: none"> <li>- Account Name as per account specified to debit.</li> </ul>
DebtorAccountNumber	S	M	19	<ul style="list-style-type: none"> <li>- The debtor's account number.</li> </ul>
DebtorBranchCode	S	M	6	<ul style="list-style-type: none"> <li>- This is the 6-digit branch code of the debtor's bank and is required to identify the bank.</li> </ul>
DebtorAccountTypeID	I	M	2	<ul style="list-style-type: none"> <li>- The debtor's type of bank account that can be selected when sending a mandate.</li> <li>- Expected values:</li> <li>- 10 = Current</li> <li>- 20 = Savings</li> <li>- 30 = Transmission</li> </ul>
DebtorIdentificationTypeID	I	M	2	<ul style="list-style-type: none"> <li>- Is the type of debtor identification that is used for the mandate.</li> <li>- Expected values:</li> <li>- 2 = ID document</li> </ul>



				<ul style="list-style-type: none"> <li>- 6 = Passport</li> <li>- 7 = Temporary residence ID</li> </ul>
DebtorIdentificationNumber	S	M	35	<ul style="list-style-type: none"> <li>- Identification number for the debtor based on the identification type selected.</li> </ul>
DebtorContactTypeID	I	M	2	<ul style="list-style-type: none"> <li>- This is the debtor's contact type for contact number or email address.</li> <li>- Expected values:</li> <li>- 2 = Office Phone</li> <li>- 11 = Email Address</li> </ul>
DebtorContactDetails	S	M	30 E-mail = 90	<ul style="list-style-type: none"> <li>- This is the value for the contact number.</li> <li>- Field may not be left blank or contain spaces between characters.</li> </ul>
MandateStatusChangeReasonCode	S	M/O	4	<ul style="list-style-type: none"> <li>- If the request is amendment or cancellation, the Mandate status change reason code must be provided.</li> <li>- The reason the mandate is being amended. Below is the description associated to each of the codes that must be provided.</li> <li>- <b>Please see table of amendment or cancellation reasons.</b></li> </ul>
BankMandateReference	S	M/O	35	<ul style="list-style-type: none"> <li>- If the request is amendment or cancellation, the Bank Mandate Reference must be provided.</li> <li>- The Bank Mandate Reference that was returned when the original mandate was accepted.</li> <li>- <b>The field cannot be amended - provide the same value as the mandate initiation. If field needs to change, new mandate required.</b></li> </ul>
OperationTypeID	I	M	1	<ul style="list-style-type: none"> <li>- The Operation Type ID states what type of operation is being done for the DebiCheck Mandate.</li> <li>- If a new collection is created, the Operation Type ID must be 1 – Initiation.</li> <li>- If a collection is updated, the Operation Type ID must be 2 – Amendment.</li> <li>- If the DebiCheck Mandate must be cancelled, the Operation Type ID must be 3 – Cancellation.</li> </ul>

## TABLE TYPES AND DEFINITIONS

### ENTRY CLASS CODES TYPES

EntryClassCodeTypeID	Description
21	Insurance Premium
22	Pension Fund Contribution
23	Medical Aid Fund Contribution
26	Unit Trust Purchase
28	Charitable or religious contribution
31	H.P Repayment
32	Account Repayment
33	Loan Repayment (other than Mortgage)

34	Rental-Lease (other than Property)
35	Service Charged (Maintenance of Service Agreements, ect.)
36	Service Charge (Variable Amounts)
37	Value Added Tax (Vat Collection)
41	Rent (Property)
42	Bond Repayment
44	Bank Use - Debit Transfer
46	Bank Use - Cheque Card Debits

#### FREQUENCY TYPES

FrequencyTypeID	Description
2	Weekly
4	Fortnightly
8	Monthly Event
16	Quarterly Event
32	Bi Annual
64	Yearly
128	Monthly By Rule

#### MANDATE STATUS TYPES

Mandate Status Types	Description
0	Awaiting Submission
1	Saved
2	Pending Authorisation
3	Accepted
4	Rejected
5	No Response
6	Internal Error
7	System Error
8	Queued

#### OPERATION TYPES

OperationTypeID	Description
1	Initiation
2	Amendment
3	Cancellation

#### MANDATE AMENDMENT REASONS

Reason Code	Description
MD16	Request By Customer
MD17	Cancellation/amendment requested By Initiating Party
MD19	Unsuspend a Mandate with changes
MD20	Unsuspend an unchanged Mandate
MS02	Reason has not been specified by End Customer

#### MANDATE CANCELLATION REASONS

Reason Code	Description
-------------	-------------

MCES	Mandate cancelled due to early settlement
CEXP	Contract expired
MCFR	Mandate cancellation due to fraud
MICN	Cancellation of a mandate initiation
MACN	Cancellation of a mandate amendment

### CREATE DEBICHECK COLLECTION

This method adds a new DebiCheck collection to an existing client. Please note that collection rules still apply. Back-end validations will also be done on DebiCheck applicable fields. The CollectionMethodTypeId must be 11 for a DebiCheck mandate.

#### REQUEST SAMPLE

URL	<i>Example: <b>Provided URL</b>/api/collections/createcollection</i>
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "clientId": 1237305,   "collectionMethodTypeId": 4,   "collectionFrequencyTypeId": 2,   "collectionStatusTypeId": 1,   "commencementDate": "2021-03-01 00:00:00",   "transactionDate": null,   "expiryDate": null,   "debitDay": 5,   "debitMonth": null,   "amount": 52,   "toCollectionAmountTotal": null,   "naedoTrackingCodeId": 12,   "collectionDescription": "Test",   "accountTypeId": 1,   "branchCode": 250655,   "accountNumber": "12345678901",   "accountName": "MK Rosewall",   "transactionDate2": null,   "transactionDate3": null,   "transactionDate4": null,   "productId": 1,   "mandate": {     "Filename": "Test.PNG",     "FileData": "0x89504E470D0A1A0A000000D494844520000011A000000DF 080600000029E11C09000000017352474200AECE1CE90000000467414D4100 00B18 F0BFC61050000000970485973000012740000127401DE661F78000 0058849444154785EEDDD8D6DD44A1840D1B4400DB4400F94400DB44007E 9800EA8800A688006E8801E160D8F41665F40F1666F98C5E7489650B4760CA 3EFCAFBE3E5EE0410131A202734404E68809CD00039A101724203E48406C80 90D90131A202734404E68809CD00039A101724203E48406C8090D90131A20</pre>

	<pre> 2734404E68809CD00039A101724203E48406C8090D90131A202734404E68... " }, "debiCheckMandate": {   "FirstCollectionDate": "2021-03-05",   "InstalmentAmount": 52.00,   "MaximumCollectionAmount": 78.00,   "InitialAmount": 52.00,   "AdjustmentCategoryTypeID": 10,   "AdjustmentAmount": 0.00,   "AdjustmentRate": 0,   "CollectionDay": 5,   "TrackingIndicator": 0,   "TransmissionNumber": 1,   "AuthenticationInstrument": "0227",   "AuthenticationType": 10,   "InitiationDate": "2021-02-23",   "InstalmentOccurrenceTypeID": 2,   "FrequencyTypeID": 8,   "EntryClassCodeTypeID": 33,   "UseCollectionDay": 1,   "DateAdjustmentAllowed": 0,   "DebtorAccountName": "MK Rosewall",   "DebtorAccountNumber": "12345678901",   "DebtorBranchCode": 250655,   "DebtorAccountTypeID": 10,   "DebtorIdentificationTypeID": 2,   "DebtorIdentificationNumber": "8905165086085",   "DebtorContactTypeID": 3,   "DebtorContactDetails": "0713352472",   "UseInitialAmount": 1,   "UseAdjustmentAmount": 0,   "UseAdjustmentRate": 0.00,   "OperationTypeID": 1,   "TransmissionTypeID": 1,   "MandateStatusChangeReasonCode": "" } } } </pre>

RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Collection ID. If it was a new Collection being created it will return the new Collection ID
Success	B	-	Indicates if the Collection was saved or not.
Messages	DTO	-	Will show a list of errors if Collection did not save successfully. Will be empty if Collection saved successfully

RESPONSE SAMPLES

SUCCESSFUL RESPONSE

Sample
<pre>{   "value": 1627208,   "success": true,   "messages": [] }</pre>

ERROR RESPONSE

Sample
<pre>{   "value": 0,   "success": false,   "messages": [     "The Collection Method Type ID '0' does not exist."   ] }</pre>

UPDATE DEBICHECK COLLECTION

This method updates DebiCheck collection. Please note that collection rules still apply. Back-end validations will also be done on DebiCheck applicable fields. The CollectionMethodTypeId must be 11 for a DebiCheck mandate. Also note that the entire collection object must be passed through. The Update DebiCheck Collection functionality will cater for mandate **amendments** and **cancellations**. For amendments and cancellations, the Mandate Status Change Reason Code and Bank Mandate Reference fields must be populated with the applicable values.

REQUEST SAMPLE

URL	<i>Example: <b>Provided URL</b>/api/collections/updatecollection</i>
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "clientId": 1237305,   "collectionMethodTypeId": 4,   "collectionFrequencyTypeId": 2,   "collectionStatusTypeId": 1,   "commencementDate": "2021-03-01 00:00:00",   "transactionDate": null,   "expiryDate": null,   "debitDay": 5,   "debitMonth": null,   "amount": 52,   "toCollectionAmountTotal": null,   "naedoTrackingCodeId": 12,   "collectionDescription": "Test",   "accountTypeId": 1,   "branchCode": 250655, }</pre>

```

"accountNumber": "12345678901",
"accountName": "MK Rosewall",
"transactionDate2": null,
"transactionDate3": null,
"transactionDate4": null,
"productId": 1,
"mandate": {
  "Filename": "Test.PNG",
  "FileData": "0x89504E470D0A1A0A0000000D494844520000011A000000DF
080600000029E11C09000000017352474200AECE1CE90000000467414D4100
00B18 F0BFC61050000000970485973000012740000127401DE661F78000
0058849444154785EEDDD8D6DD44A1840D1B4400DB4400F94400DB44007E
9800EA8800A688006E8801E160D8F41665F40F1666F98C5E7489650B4760CA
3EFCAFBE3E5EE0410131A202734404E68809CD00039A101724203E48406C80
90D90131A202734404E68809CD00039A101724203E48406C8090D90131A20
2734404E68809CD00039A101724203E48406C8090D90131A202734404E68...
"
},
"debiCheckMandate": {
  "FirstCollectionDate": "2021-03-05",
  "InstalmentAmount": 52.00,
  "MaximumCollectionAmount": 78.00,
  "InitialAmount": 52.00,
  "AdjustmentCategoryTypeID": 10,
  "AdjustmentAmount": 0.00,
  "AdjustmentRate": 0,
  "CollectionDay": 5,
  "TrackingIndicator": 0,
  "TransmissionNumber": 1,
  "AuthenticationInstrument": "0227",
  "AuthenticationType": 10,
  "InitiationDate": "2021-02-23",
  "InstalmentOccurrenceTypeID": 2,
  "FrequencyTypeID": 8,
  "EntryClassCodeTypeID": 33,
  "UseCollectionDay": 1,
  "DateAdjustmentAllowed": 0,
  "DebtorAccountName": "MK Rosewall",
  "DebtorAccountNumber": "12345678901",
  "DebtorBranchCode": 250655,
  "DebtorAccountTypeID": 10,
  "DebtorIdentificationTypeID": 2,
  "DebtorIdentificationNumber": "8905165086085",
  "DebtorContactTypeID": 3,
  "DebtorContactDetails": "0713352472",
  "UseInitialAmount": 1,
  "UseAdjustmentAmount": 0,
  "UseAdjustmentRate": 0.00,
  "OperationTypeID": 2,
  "TransmissionTypeID": 1,
  "MandateStatusChangeReasonCode": "MD16"
}
}
}

```

--	--

RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Collection ID. If it was a new Collection being created it will return the new Collection ID
Success	B	-	Indicates if the Collection was saved or not.
Messages	DTO	-	Will show a list of errors if Collection did not save successfully. Will be empty if Collection saved successfully

RESPONSE SAMPLES

SUCCESSFUL RESPONSE

Sample
<pre>{   "value": 1627208,   "success": true,   "messages": [] }</pre>

ERROR RESPONSE

Sample
<pre>{   "value": 0,   "success": false,   "messages": [     "The Collection Method Type ID '0' does not exist."   ] }</pre>

MANDATES

MANDATE DOWNLOAD

REQUEST SAMPLE

URL	<p><b>Provided</b> URL/api/mandates/downloadmandate/{DocumentGuid}  <i>Example: <b>Provided</b> URL/api/mandates/downloadmandate/16233E33-ECE2-4A5C-9852-65C2BD27805D</i></p>
-----	---

Method	GET
Headers	Content-Type: <b>application/octet-stream</b> Authorization: Bearer <b>Generated Bearer Token</b>

---

RESPONSE SAMPLE

Sample
<pre> "0x255044462D312E350D0A25B5B5B5B50D0A312030206F626A0D0A3C3C2F547970652F43617461 6C6F672F50616765732032203020522F4C616E6728656E2D555329203E3E0D0A656E646F626A0D0A 322030206F626A0D0A3C3C2F547970652F50616765732F436F756E7420312F4B6964735B20332030 20525D203E3E0D0A656E646F626A0D0A332030206F626A0D0A3C3C2F547970652F506167652F506 172656E742032203020522F5265736F75726365733C3C2F466F6E743C3C2F46312035203020522F46 32203131203020523E3E2F4578744753746174653C3C2F47533130203130203020523E3E2F50726F6 35365745B2F5044462F546578742F496D616765422F496D616765432F496D616765495D203E3E2F4 D65646961426F785B2030203020363132203739325D202F436F6E74656E74732034203020522F477 26F75703C3C2F547970652F47726F75702F532F5472616E73706172656E63792F43532F4465766963 655247423E3E2F546162732F533E3E0D0A656E646F626A0D0A342030206F626A0D0A3C3C2F46696 C7465722F466C6174654465636F64...." </pre>

You will need to handle the file byte[] returned on your side.

## PRODUCTS

Field Name	Type	M/O/C	Length	Description
Product	DTO		-	Product Data Transfer Object
ProductID	N	M	-	The system ID for the product. Will be used on other requests to uniquely reference the product.
Name	S	M	50	Your Product or Service name.
Description	S	M	250	A description of the product or service you provide.
Price	N	M	-	The default price of your product or service. When a new Collection is added, the amount column will be set to the price specified. If a Collection is edited, the user has the option of leaving the amount unchanged, or setting the amount as per the Product or Service price.
CanEdit	B	M	-	If Can Edit is set to true (yes), you will be able to adjust the amount on the Collections Add or Edit screens/Api call as per Product or Service selected/specified. If set to false (no), you won't be able to adjust the amount. The amount will be set as per the selected Product or Service price.



Active	B	M	-	Will indicate if the Product or Service is currently active or not. Please note: If set to inactive, the Collections with this Product or Service reference will remain active
IsDefault	B	M	-	Your default Product or Service. When a new Collection is added, the default Product or Service will automatically be selected. Only one Product or Service can be set as the default.
Reference	S	M	8	The unique reference for the Product or Service created.

---

## PRODUCT FIELDS

### GET PRODUCT LIST

This method allows the user to list all products on their profile.

---

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/products/listproducts <i>Example: <b>Provided URL</b>/api/products/listproducts</i>
Method	GET
Headers	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

---

### RESPONSE SAMPLE

Sample
<pre>[   {     "productId": 2,     "name": "Test 1",     "description": "Default Product/Service",     "price": 0.00,     "canEdit": true,     "active": true,     "isDefault": true,     "reference": "SOFTY00000"   },   {     "productId": 1573,     "name": "Test 2",     "description": "testing product",     "price": 50.00,     "canEdit": false,     "active": true,</pre>

```

    "isDefault": false,
    "reference": "SOFTY00001"
  }
]

```

## GET PRODUCT DETAILS

This method allows the user to get a specific product on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/products/getproductdetails/{productID} <i>Example: <b>Provided URL</b>/api/products/getproductdetails/1573</i>
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>

### RESPONSE SAMPLE

Sample
<pre> {   "productId": 1573,   "name": "Test",   "description": "testing product",   "price": 50.00,   "canEdit": false,   "active": true,   "isDefault": false,   "reference": "SOFTY00000" } </pre>

## INSERT PRODUCT

This method allows the user to add a new product on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/products/createproduct
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre> {   "productId": 0, </pre>

```

"name": "Test 2",
"description": "new testing product for testing",
"price": 50.00,
"canEdit": false,
"active": true,
"isDefault": false,
"reference": "SOFTY00000"
}

```

RESPONSE FIELDS

Field Name	Type	Length	Description
Value	N	-	This will contain the Product ID. If it was a new product being created it will return the new Product ID
Success	B	-	Indicates if the Product was saved or not
Messages	DTO	-	Will show a list of errors if Product did not save successfully. Will be empty if product saved successfully

RESPONSE SAMPLES

SUCCESSFUL RESPONSE

```

Sample
{
  "value": 1574,
  "success": true,
  "messages": []
}

```

ERROR RESPONSE

```

Sample
{
  "value": 0,
  "success": false,
  "messages": [
    "Please check that your request object is correct"
  ]
}

```

## UPDATE PRODUCT

This method allows the user to update a specific client on their profile.

### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/products/updateproduct
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	<pre>{   "productId": 1573,   "name": "Test",   "description": "testing product for testing",   "price": 50.00,   "canEdit": false,   "active": true,   "isDefault": false,   "reference": "SOFTY00000" }</pre>

### RESPONSE FIELDS

Field Name	Type	Length	Description
value	N	-	This will contain the Product ID. If it was a new client being created it will return the new Client ID
success	B	-	Indicates if the Product was saved or not
messages	DTO	-	Will show a list of errors if Product did not save successfully. Will be empty if Product saved successfully.

### RESPONSE SAMPLE

#### SUCCESSFUL RESPONSE

Sample
<pre>{   "value": 1573,   "success": true,   "messages": [] }</pre>

---

ERROR RESPONSE

Sample
<pre>{   "value": 2,   "success": false,   "messages": [     "Product with ID 2 not found."   ] }</pre>

**TRANSACTION OPERATIONS**

**TRANSACTIONS**

---

TRANSACTION FIELDS

Field Name	Type	Length	Description
TransactionID	N	-	The primary key.
CollectionID	N	-	The collection id the transaction is linked to.
ClientID	N	-	The client id the collection belongs to.
TransactionTypeID	N	-	The type of transaction.
TransactionDate	DT	-	The date of the transaction.
Amount	N	-	The amount.
RunningBalance	N	-	The running balance of the collection at the moment of the transaction.
TotalRunningBalance	N	-	The total running balance of the client transactions at the moment of the transaction.
Reference	S	19	The reference on the client bank statement generated by the system.
GenerationTypeID	N	-	The type of transaction for the generation. Normal, Arrear, Unpaid, Resubmission.
CycleDate	DT	-	The cycle data to the original transaction. NAEDO applicable.

---

GET TRANSACTION LIST BY COLLECTIONID

This method allows the user to retrieve a list of transaction history for a specific collection.

---

REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
CollectionID	N	M	-	The collection id that the history is searched for.

---

 REQUEST SAMPLE

URL	<b>Provided URL</b> /api/transactions/listtransactionsbycollection/{collectionID} <i>Example: <b>Provided URL</b>/api/transactions/listtransactions/164061</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>

---

 RESPONSE SAMPLE

Sample
<pre>[   {     "transactionId": 1081898,     "transactionTypeId": 4,     "clientId": 922,     "transactionDate": "2015-02-06T00:00:00",     "insertionDate": "2015-02-04T00:00:00",     "amount": 724.19,     "runningBalance": 0,     "collectionId": 164061,     "collectionGenId": null,     "reference": "SOFTY 000001  ",     "sequenceNumber": 4686,     "totalRunningBalance": 0,     "originalTransactionId": 1081881,     "fileSequenceNo": null,     "transactionSubmitted": 1,     "generationTypeId": 1,     "cycleDate": "2015-02-06T00:00:00",     "client": null,     "generationType": null,     "transactionType": null,     "unpaysData": []   },   {     "transactionId": 1081881,     "transactionTypeId": 4,     "clientId": 922,     "transactionDate": "2015-02-06T00:00:00",     "insertionDate": "2015-02-04T00:00:00",     "amount": -724.19,     "runningBalance": -724.19,     "collectionId": 164061,</pre>

```

    "collectionGenId": 1777,
    "reference": "SOFTY 000001",
    "sequenceNumber": null,
    "totalRunningBalance": -724.19,
    "originalTransactionId": null,
    "fileSequenceNo": null,
    "transactionSubmitted": 1,
    "generationType": 1,
    "cycleDate": "2015-02-06T00:00:00",
    "client": null,
    "generationType": null,
    "transactionType": null,
    "unpaysData": []
  }
]

```

---

#### GET TRANSACTION LIST BY TRANSACTION DATE

This method allows the user to retrieve a list of transaction history for a specific collection.

---

#### REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
TransactionDate	DT	M	-	The applicable Transaction Date.

---

#### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/transactions/listtransactions/{transactionDate} <i>Example: <b>Provided URL</b> /api/transactions/listtransactionsbydate/2015-02-06</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>

---

#### RESPONSE SAMPLE

Sample
<pre> [   {     "transactionId": 1081881,     "transactionType": 4,     "clientId": 922,     "transactionDate": "2015-02-06T00:00:00",     "insertionDate": "2015-02-04T00:00:00",     "amount": -724.19,     "runningBalance": -724.19,     "collectionId": 164061,     "collectionGenId": 1777,     "reference": "SOFTY 000001", </pre>

```

"sequenceNumber": null,
"totalRunningBalance": -724.19,
"originalTransactionId": null,
"fileSequenceNo": null,
"transactionSubmitted": 1,
"generationTypeld": 1,
"cycleDate": "2015-02-06T00:00:00",
"client": null,
"generationType": null,
"transactionType": null,
"unpaysData": []
},
{
  "transactionId": 1081880,
  "transactionTypeld": 4,
  "clientId": 47553,
  "transactionDate": "2015-02-06T00:00:00",
  "insertionDate": "2015-02-04T00:00:00",
  "amount": -1200,
  "runningBalance": -1200,
  "collectionId": 50442,
  "collectionGenId": 1777,
  "reference": "VULAMEHL000008",
  "sequenceNumber": null,
  "totalRunningBalance": -1200,
  "originalTransactionId": null,
  "fileSequenceNo": null,
  "transactionSubmitted": 1,
  "generationTypeld": 1,
  "cycleDate": "2015-02-06T00:00:00",
  "client": null,
  "generationType": null,
  "transactionType": null,
  "unpaysData": []
}
]

```

---

#### GET TRANSACTION LIST BY TRANSACTION DATE RANGE

This method allows the user to retrieve a list of transaction history for a specific date range.

---

#### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/transactions/listtransactionsbydateRange <i>Example: <b>Provided URL</b>/api/transactions/listtransactionsbydateRange</i>
Method	GET OR POST
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>
Request	{ "FromDate": "2020-09-01" "ToDate": "2020-10-31"



```
    }
```

#### RESPONSE SAMPLE

Sample
<pre>[   {     "insertionDate": "2015-02-04T00:00:00",     "amount": -1200,     "runningBalance": -1200,     "collectionId": 50442,     "collectionGenId": 1777,     "reference": "VULAMEHL000008",     "sequenceNumber": null,     "totalRunningBalance": -1200,     "originalTransactionId": null,     "fileSequenceNo": null,     "transactionSubmitted": 1,     "generationTypeId": 1,     "cycleDate": "2015-02-06T00:00:00",     "client": null,     "generationType": null,     "transactionType": null,     "unpaysData": []   } ]</pre>

#### UNPAIDS

##### UNPAID TRANSACTION FIELDS

Field Name	Type	Length	Description
TransactionID	N	-	The primary key.
CollectionID	N	-	The collection id the transaction is linked to.
ClientID	N	-	The client id the collection belongs to.
TransactionTypeID	N	-	The type of transaction.
TransactionDate	DT	-	The date of the transaction.
Amount	N	-	The amount.
RunningBalance	N	-	The running balance of the collection at the moment of the transaction.
TotalRunningBalance	N	-	The total running balance of the client transactions at the moment of the transaction.
Reference	S	19	The reference on the client bank statement generated by the system.
GenerationTypeID	N	-	The type of transaction for the generation. Normal, Arrear, Unpaid, Resubmission.
CycleDate	DT	-	The cycle data to the original transaction. NAEDO applicable.

UnpaidReasonDescription	S	100	Reason for the unpaid.
-------------------------	---	-----	------------------------

---

#### GET UNPAIDS BY COLLECTION

This method will return all unpaid transaction within the given date range.

---

#### REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
CollectionID	N	M	-	The applicable collection ID.

---

#### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/transactions/listunpaidbycollection/{collectionID} <i>Example: <b>Provided URL</b> /api/transactions/listunpaidbycollection/609887</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>

---

#### RESPONSE SAMPLE

Sample
<pre>[   {     "unpaidReasonDescription": "Not Provided For",     "transactionId": 4846331,     "collectionId": 609887,     "clientId": 444865,     "transactionTypeId": 7,     "transactionDate": "2018-05-16T00:00:00",     "amount": -1180.46,     "runningBalance": -1180.46,     "totalRunningBalance": -1180.46,     "reference": "SOFTY 000035",     "generationTypeId": 1,     "cycleDate": "2018-05-15T00:00:00"   } ]</pre>

---

**GET UNPAIDS BY TRANSACTION DATE**

This method will return all unpaid transaction within the given transaction date.

---

**REQUEST FIELDS**

Field Name	Type	M/O/C	Length	Description
transactionDate	DT	M	-	The date of the transaction.

---

**REQUEST SAMPLE**

URL	<b>Provided URL</b> /api/transactions/listunpaidbydate/{transactionDate} <i>Example: <b>Provided URL</b> /api/transactions/listunpaidbydate/2018-05-16</i>
Method	GET
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>

---

**RESPONSE SAMPLE**

Sample
<pre>[   {     "unpaidReasonDescription": "Not Provided For",     "transactionId": 4846331,     "collectionId": 609887,     "clientId": 444865,     "transactionTypeId": 7,     "transactionDate": "2018-05-16T00:00:00",     "amount": -1180.46,     "runningBalance": -1180.46,     "totalRunningBalance": -1180.46,     "reference": "SOFTY 000035",     "generationTypeId": 1,     "cycleDate": "2018-05-15T00:00:00"   } ]</pre>

---

**GET UNPAIDS BY TRANSACTION DATE RANGE**

This method will return all unpaid transaction within the given date range.

REQUEST SAMPLE

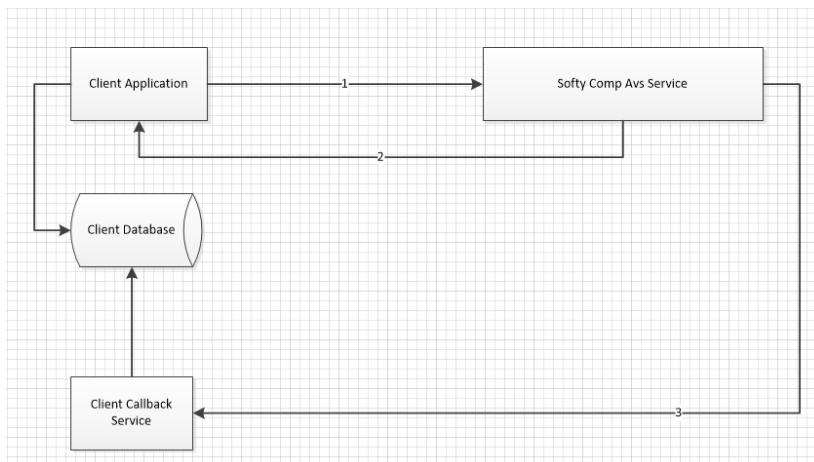
URL	<b>Provided URL</b> /api/transactions/listunpaidbydaterange <i>Example: <b>Provided URL</b>/api/transactions/listunpaidbydaterange</i>
Method	GET OR POST
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>
Request	{ "FromDate": "2020-09-01" "ToDate": "2020-10-31" }

RESPONSE SAMPLE

Sample
[ { "unpaidReasonDescription": "Not Provided For", "transactionId": 4846331, "collectionId": 609887, "clientId": 444865, "transactionTypeId": 7, "transactionDate": "2018-05-16T00:00:00", "amount": -1180.46, "runningBalance": -1180.46, "cycleDate": "2018-05-15T00:00:00" } ]

AVS OPERATIONS

REQUEST AVS RECORD



This method allows the user to do an AVS check. Please take note that you will be billed per request. Contact Softy Comp for billing and pricing enquiries.

With the AVS you will be required to host a call back service that will receive the response once they are processed and responses are received. Softy Comp will push the response of the request to the call back service via the end-user's choice of communication (SOAP or JSON).

Real-time participants (± 3 minute. responses):

- If real-time channel problems occur, resubmissions occur automatically through Normal banks channel, with delayed responses (± 1h 15m).
- Requests can be made any time.

Normal participants (± 1 hour responses):

- Requests can be made between 3:10 and 16:00. After hour requests receive responses at 3:00.

Participating Banks:

	Bank	Avs Normal	AVS Real Time
1	Standard Bank	✓	✓
2	ABSA	✓	✓
3	Nedbank	✓	✓
4	FNB	✓	✓
5	Capitec	✓	✓
6	African Bank	✓	✓
7	Investec	✓	✓
8	Discovery	✓	✓
9	Sasfin	✗	✓
10	Mercantile	✓	✗
11	Grobank (Old Bank of Athens)	✓	✗
12	Bidvest	✓	✗
13	Grindrod	✓	✓

REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
BranchCode	S	M	6	The account number branch code.
AccountNumber	S	M	21	The account number.
AccountTypeID	I	M	-	The account type id.

IDNumber	S	M	13	The ID Number for the account.
Initials	S	M	3	The initials.
Name	S	M	30	The account holder name/surname.
EmailAddress	S	O	60	The account holder email address.
CellNo	S	O	10	The account holder cell phone number.
CallBackUrl	S	M	100	The call back URL to where the results for the record will be pushed to when it comes through.

---

**RESPONSE FIELDS**

Field Name	Type	Length	Description
Value	S	-	Will contain the system reference that must be used to match the AVS result when pushed back to the client via the callback url provided by the client.
Success	B	-	Will indicate if AVS was successfully submitted to Soft Comp.
Messages	DTO	-	This will contain a list any error messages that occurred during the AVS Request to Softy Comp.

---

**REQUEST AND RESPONSE SAMPLE**

URL	<b>Provided URL</b> /api/avs/requestAvsAuthentication
Method	POST
Header	Content-Type: application/json Authorization: Bearer <b>Bearer Token Generated</b>
Request	{ "BranchCode": "012345", "AccountNumber": "0123456789", "AccountTypeID": 1, "IDNumber": "9010305027087", "Initials": "J", "Name": "Doe", "EmailAddress": "Roelof@softycomp.co.za", "CellNo": "0722732166", }

	<pre>"CallBackUrl": "https://scoretest.softycomp.co.za/SoftyCompBureauCallbackWCF/SoftyCompBureauCallback.WCF.SoftyCompBureauCallbackService.svc" }</pre>
Success Response Example	<pre>{ "value": "1AVS201907111522598876", "success": true, "messages": [] }</pre>
Error Response Example	<pre>{ "value": "", "success": false, "messages": [ "The provided call back url is not valid!" ] }</pre>

## CLIENT AVS CALLBACK IMPLEMENTATION GUIDE

### EXAMPLE PROJECT



avsrealtimeclientService.zip

The client must implement the "AcceptAvsResponseObjectV2" method (See [Example Project](#)) on their side. This can be done in any coding language as long as the url is made available to us. Responses will be sent back as a JSON string but please keep in mind that we can also send the client a SOAP response if that is what they prefer although our preference is JSON.

We will attempt to send the avs response 21 times to the client. If all 21 responses failed to reach the client the response submissions will time out and no further submissions will be attempted. The implemented call back method must send back a Boolean value once the response is received.

Should you require this to be submitted again later (i.e. service came back online again), you can contact us.

If the call back method returns false continuously, or any other error, the following re-send delays will be applied:

1. First 5 Times: 1 minute
2. Next 5 Times: 10 minutes
3. Next 5 Times: 30 minutes
4. Next 5 Times: 1 hour
5. Last try: 24 hours
6. Example Of how **we** would push the data back to the client:

URL	Client Provided URL as in AVS Request/AcceptAvsResponseObjectV2
-----	---

	<i>Example: https://helloworld.co.za/CallbackWCF/AcceptAvsResponseObjectV2</i>
Method	POST
Header	Content-Type: application/json
Request	<pre>{   "Message": "Response Received",   "SystemReference": "135AVS201907260145311433",   "ResponseCode": "0",   "AccountFound": "Yes",   "AccountOpen": "Yes",   "AccountTypeMatch": "Yes",   "IDMatch": "Yes",   "InitMatch": "Yes",   "NameMatch": "Yes",   "AccountAcceptDebits": "Yes",   "AccountAcceptCredits": "Yes",   "AccountLengthMatch": "Yes",   "ResponseDateTime": "2019-07-26 01:46:34.500",   "BankName": "CAPITEC BANK LTD",   "BranchName": "CAPITEC BANK CPC",   "CellNoMatch": "Yes",   "EmailAddressMatch": "Yes" }</pre>
Client Success Response Example	true
Client Error Response Example	false

## CDV (CHECK DIGIT VERIFICATION)

### REQUEST FIELDS

Field Name	Type	M/O/C	Length	Description
BranchCode	S	M	6	The account number branch code.
AccountNumber	S	M	21	The account number.
AccountTypeID	I	M	-	The account type id.

### RESPONSE FIELDS



Field Name	Type	Length	Description
Code	N	6	The code related to an Error if CDV Failed, if CDV check passed this will be 0.
Message	S	max	Message related to the result.
Success	B	-	Indicates if CDV passed or not.

---

#### REQUEST SAMPLE

URL	<b>Provided URL</b> /api/avs/requestCdv <i>Example: <b>Provided URL</b>/api/avs/requestcdv</i>
Method	POST
Headers	Content-Type: application/json Authorization: Bearer <b>Generated Bearer Token</b>
Request	{ "AccountNumber": "0123456789", "BranchCode": 12345, "AccountTypeID": 1 }
Success Response Example	{ "code": 0, "message": "Account passed CDV.", "success": true }
Error Response Example	{ "code": 95, "message": "Account Failed Check Digit Verification.", "success": true }

#### TABLE TYPES AND DEFINITIONS

---

##### CLIENT STATUS TYPES

ID	Description
1	Inactive
2	Active
3	Awaiting Approval

---

##### CLIENT TYPES

ID	Description
1	Company
2	Individual

---

##### ADDRESS TYPES

ID	Description
-1	Not Specified
1	Street
2	Postal
3	P/Bag
4	Post net Suite

---

COLLECTION STATUS TYPES

ID	Description
1	Active
2	Inactive
3	Expired
4	Awaiting AVS Approval
5	AVS Failed
6	Arrears
7	Suspended
8	Awaiting Mandate Upload

---

COLLECTION FREQUENCY TYPES

ID	Description
1	Once-Off
2	Monthly
3	Yearly
4	To Collect Amount
5	2x Once-Off
6	4x Once-Off
7	Weekly
8	Fortnightly
9	Quarterly
10	Bi-Annually
11	Adhoc

---

COLLECTION METHOD TYPES

ID	Description
1	Cash
2	Cheque
3	Credit Card
4	EFT
5	Naedo
11	DebiCheck Naedo
12	DebiCheck EFT

---

ACCOUNT TYPES

ID	Description
1	Cheque

2	Savings
3	Transmission
4	Bond
5	N/A
6	Subscription/Share

---

NAEDO TRACKING CODES

ID	Description
12	0 Days
13	1 Day
14	3 Days
15	7 Days
16	14 Days
17	21 Days
18	32 Days
19	2 Days
20	4 Days
21	5 Days
22	6 Days
23	8 Days
24	9 Days
25	10 Days

---

GENERATION TYPES

ID	Description
1	Normal
2	Arrear
3	Resubmission
4	Manual Entry
5	System Entry

---

TRANSACTION TYPES

ID	Description
1	Cash
2	Cheque
3	Credit Card
4	Debit Order
5	NAEDO
6	Home back
7	Unpaid
8	Arrear Collection
9	Credit Note
10	Negative Entry
11	DebiCheck Naedo
12	DebiCheck Debit Order
13	Refund

---

UNPAID REASON TYPES

Type Description
Account Closed
Account Effects Not Cleared
Account Frozen
Account Holder Deceased
ACCOUNT HOLDER REQUESTED BLOCK
Account In Advance
Account In Liquidation
Account In Sequestration
Account Open - Not Paid Out
Account transferred To Other Bank Group
Account Transferred Within Bank Group
Authorisation Cancelled
Balance Exceeds Maximum
Bond Cancelled
Bridges Error
CDV FAILURE
Client Did Not Authorise Debit
Closed ACB Branch Code
Closed Beneficiary Code
Credits Only. Not Unique Transaction Reference
Credits only. Unrecognised Initiating Party
Data Base Down
Debit Contravenes Client's Authority
DEBITS ARE NOT ALLOWED FOR THIS BRANCH
Debits/Credits Not Allowed
Distribution Upfront Rejection
Exceptions Error
History Record Not Found
HOMING ACC. NO FAILS CDV EXCEPTION CODE
HOMING ACC. NO FAILS CDV VERIFICATION
HOMING ACC. NO IS ZERO
HOMING ACC. NO NO MATCHING ACCT TYPE CDV FOUND
HOMING ACC. NO NO MATCHING ACCTYPE CDV FOUND
HOMING ACC.NAMEIS BLANK
HOMING BRANCH IS NON-EXISTENT
Interest Calculation Error
Interest/Capital Exceeded
Invalid Branch
Invalid Data
Invalid Mode
Limits Violation

New Mortgage Loan Financial Error
New Mortgage Loans Invalid Due Date
No Book Error
No Such Account
No Transfer, Account In Advance
Not FICA Compliant
Not Provided For
Old/New Balances Differ
Online Transaction In Progress
ONLY CR. TRANS-ACTIONS AGAINSTA SAVINGS ACC
Original Tran Not Found
OTR Error/Refer EPSQ History
Override Required
PAAF Insufficient Funds
Payment Stopped
Payment stopped by account holder
Possible Stop Payment
Post Dated Transaction
Previously Stopped As Stop Payment
Recall/Withdrawal
Subscription Amount Required
Successful
Tran Backdated Beyond Limit
Tran on Managers Referral
Transaction Withdrawal
UNPROCESSED BY BANK - PLEASE RESUBMIT