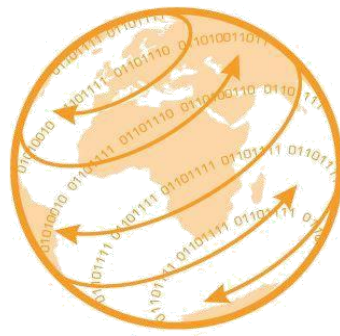




eCert Web API User Guide

13 August 2020

Table of Contents

13 AUGUST 2020	0
PURPOSE OF DOCUMENT	2
WEB API VERSIONS	2
PRE-REQUISITES.....	2
HOW TO CONNECT	3
TESTING WEB API ENDPOINTS	3
SWAGGER	3
AVAILABLE METHODS	3
1. APPLY FOR EPHYTO CERTIFICATE (eCERT): POST	3
1.1 <i>Update ePhyto Certificate (eCert)-POST</i>	12
2. GET CERTIFICATE DATA (eCERT): GET	13
3. GET CERTIFICATE PDF (eCERT): GET	13
4. GET ADDITIONAL INPUT (eCERT): GET	14
5. POST ADDITIONAL INPUT (eCERT): POST	15
6. CHECK PHYTO APPLICATION STATUS (eCERT): GET	15
7. ADD DOCUMENTS (eCERT):POST.....	17
8. GET LIST OF AGREEMENTS (eCERT): GET	18
9. GET LIST OF AGREEMENTS BY CODE (eCERT): GET.....	20
10. GET DATA FOR STANDARD (eCERT): GET.....	21
11. LIST OF STANDARDS (eCERT): GET	22
API CHANGE LOG	23
APPENDIX 1	25
REFERENCE DOCUMENTS	25
WEB REFERENCES.....	25

Purpose of Document

This document demonstrates the step by step process of setting up and using the eCert Web API methods to perform an application for an electronic Phytosanitary certificate from the request to the issuance of the certificate. To interact with the eCert Web API, client systems need to be registered.

Web API Versions

There are three versions of the API that are available at a time and these are stated below:

- **Old** - Soon to be deprecated
- **Stable** – Current stable version
- **Beta** - Latest Build

All requests are sent with the version of the API in the URL or the consumer can call the API through a custom request header as below:

```
Accept-version: v1  
Accept-version: v2
```

The current version of the API will be displayed on the API's Swagger page. The swagger page will display the available endpoints depending on the selected version.

When a new version becomes available, notifications will be sent out to all the relevant parties with a changelog of the new changes and features. The Swagger page will be updated accordingly with these new features. The links to the swagger pages are as follows:

- PRODUCTION - <https://app.ecert.co.za/api/>
- TESTING- <http://qa.ecert.co.za/api/>

Pre-requisites

The pre-conditions for a client system to be able to consume the eCert Web API for ePhyto requests are as follows:

- The client system must belong or be associated with a registered business in the Central Business Register
- The client system will need to be registered as a client in the User Authentication Service and issued a valid **client_id** and **client_secret** to be able to consume the endpoints
- All request calls for the eCert Notice will be authenticated using a JWT (JSON Web token) bearer token over OAuth 2.0 protocol
- The authorization grant type used by the User Authentication Service is 'Client Credentials Grant' (please see Appendix 1)

How to Connect

Refer to the **How to Connect to Web API Document**

Testing Web API Endpoints

To test the eCert Web API endpoints users can use 3rd party tools like Swagger or Postman. These tools will help the user get a feel of what the endpoint parameters look like, as well as view the responses in various formats. Following is the endpoint to the QA eCert API to see the available standards:

Swagger

- <http://qa.ecert.co.za/api/swagger> (TESTING)

Available Methods

1. Apply for ePhyto Certificate (eCert): POST

PLEASE NOTE: this is currently reserved for Fruit.

Used to apply for ePhyto certificate by submitting all the necessary information to process the application.

Endpoint:

- <https://app.ecert.co.za/api/v1/certificate/apply> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/certificate/apply> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/ Ref
-CertificateRequest				
ApplicationReferenceNo	Application Reference Number	String	No	e.g. NPPO-ZA/2020/4/25001
ApplicationType	The type of application	Int	Yes	e.g. 851 (for Phytosanitary Certificate) or 657 (for Reexport)
ApplicationStatus	The status of the application	String	Yes	e.g. P (for Provisional), S (for Submit), U (for Update)

CBRID	The owner of the consignment	Int	Yes	e.g. 3 Central Business Register
CBRBillingID	The responsible person to bill	Int	Yes	e.g. 45 Central Business Register
CustomReferenceNo	Sent by the consumer to reference their system	String	No	e.g. 2016044
AgreementCode	The agreement code	String	Yes	e.g. AGM0001 Agreements
DesiredIssueLocation	The location where Phyto will be issued	String	Yes	e.g. Durban, Cape Town
NotificationEmail	Email address for the individual to be notified of progress	String	No	e.g. bob@fruit.co.za
ConsignorId	Exporter code	String	No	e.g. D9999
ConsignorName	Name of exporter	String	Yes	e.g. Overlake Farm
ConsignorAddressLine1	Exporter address line 1	String	Yes	
ConsignorAddressLine2	Exporter address line 2	String	No	
ConsignorAddressLine3	Exporter address line 3	String	No	
ConsigneeId	The party importing the consignment's business code	String	No	e.g. VDLANS
ConsigneeName	Consignee name	String	Yes	e.g. VAN DER LANS INTERNATIONAL B.V.
ConsigneeAddressLine1	Consignee address line 1	String	Yes	
ConsigneeAddressLine2	Consignee address line 2	String	No	
ConsigneeAddressLine3	Consignee address line 3	String	No	

ConsigneeCountryId	Consignee's country	Lookup	Yes	e.g. "NL" Country ISO Alpha-2 Code
ImportCountryId	Importing Country	Lookup	Yes	e.g. "NL" Country ISO Alpha-2 Code
TargetCountryId	The country where the goods are being delivered to	Lookup	Yes	e.g. "NL" Country ISO Alpha-2 Code
UnloadingBaseportLocation	The final port of consignment unload	Lookup	Yes	e.g. RTM Locations
TransportModeCode	Type of transport used	Lookup	Yes	e.g. 1 (Maritime Transport) Modes of Transport
TransportMeans	The means of transport	String	No	e.g. Container Vessel
PaymentType	The method of payment to be used	String	No	e.g. Cash or Account
>> ConsignmentItems				
>>> ConsignmentItem				
Description	Product Description	String	No	e.g. Crimson Seeds
CommonName	General product name	String	Yes	e.g. Mandarins Standards = ScientificNameOfPlant
ScientificName	The scientific name of the product	String		e.g. <i>Citrus Sinensis</i>
NetWeightMeasureCode	Unit of measure for net weight	Lookup	Yes	e.g. KGM Units of Measures Codes
NetWeightMeasure	Net weight	Decimal	Yes	e.g. 23315.234
GrossWeightMeasureCode	Unit of measure for the gross weight	String	Yes	e.g. KGM Units of Measures Codes
GrossWeightMeasure	Gross weight	Decimal	Yes	e.g. 23320.234
CustomsHarmonizedSystemClass	Harmonized system code	String	No	e.g. 000000

CommodityVegetablePart	Type of commodity class	Lookup	No	e.g. Fruits Commodity Class Codes
CommodityConditionClass	Condition of commodity	Lookup	No	e.g. Fresh Commodity Conditions
CommodityIntentOfUseClass	Commodity's intent of use	Lookup	No	e.g. Consumption Commodity Intent Of Use List
AppliedProcessStartDate	Treatment Start Date	String	No	W3 Format YYYY-MM-DDThh:mmTZD (1997-07-16T19:20+01:00)
AppliedProcessEndDate	Treatment End Date	String	No	W3 Format YYYY-MM-DDThh:mmTZD (1997-07-16T19:20+01:00)
DurationMeasureCode	Unit code for the duration of treatment	String	No	e.g. H – Hours or D – Days
DurationMeasure	Duration of treatment	Decimal	No	e.g. 3
AppliedProcessTreatmentTypeLevel1	Treatment Type Level 1	Lookup	No	e.g. TPT Treatment Types
AppliedProcessTreatmentTypeLevel2	Treatment Type Level 2	Lookup	No	e.g. CT Treatment Types
AppliedProcessChemicalCode	Chemical Ingredient	Lookup	No	e.g. Clothianidin Active Ingredient List
AppliedProcessTemperatureUnitCode	Unit of measure for temperature	Lookup	No	e.g. HUR Units of Measures Codes
AppliedProcessTemperature	Temperature applied	Decimal	No	e.g. 0.6
AppliedProcessConcentrationUnitCode	Unit of measure for concentration	String	No	e.g. GM Units of Measures Codes
AppliedProcessConcentration	Concentration Level	Decimal	No	e.g. 2.5
AppliedProcessAdditionalNote	Additional notes for treatments	String	No	

FullTreatmentInformation	Summary of the full treatment as would appear on the Phyto. (Free-text field used to capture the entire treatment information)	String	No	e.g. Date: 2011-11-30; Type:Chemical; Duration: 3 h; Temperature: 32 C; Concentration: 2.5 ml/kg; AdditionalInformation: No additional Information available
PackageLevelCode	Package Level	Int	No	e.g. Default = 1
PackageTypeCode	Package Type	Lookup	Yes	e.g. CT Packaging Codes
PackageItemUnitCode	Unit of code to quantify the packaged item	Lookup	Yes	e.g. H87 (Default) Units of Measures Codes
PackageItemQuantity	Quantity of packaged items	Int	Yes	e.g. 1435
PackageShippingMarks	Visible Shipping Marks	String	Yes	e.g. NONE (default) if there are no codes, or e.g. SZLU96455026
AdditionalConsignmentNotes	Additional notes or declarations for consignment	String	No	e.g. The shipment was inspected and found free of Trogoderma variable.
SequenceNumeric	A number used to uniquely identify a consignment for updates	Int	Yes	e.g 1
<<< ConsignmentItem				
<< ConsignmentItems				
>> AdditionalInformation				
LoadSeaPoint	Loading Sea Port	String	Yes	e.g. ZACPT
TargetCountry	The country where the	Lookup	Yes	e.g. Netherlands

	goods are being delivered to			
TargetRegion	The region where the goods are being delivered to	Lookup	Yes	e.g. Gelderland
PortOfEntry	First Port of Entry	String	Yes	e.g. RTM Standards = Location
BookingRef	Booking Reference	String	No	e.g. 032CAL1617983
Vessel	Name of the vessel shipping	String	No	e.g. MSC Sofia Celeste
VoyageNumber	Voyager Number	String	No	e.g. NZ738R
DepartureDate	Date of departure	String	Yes	W3 Format YYYY-MM-DD
TemperatureRegime	Temperature Regime	String	No	e.g. EW001
>>> TradeUnits				
>>>> TradeUnit				
TradeUnitID	Trade Unit's unique identifier	String	Yes	e.g. GS1 – SSCC format
ClientRef	Client reference for trade unit	String	No	e.g. Z129
PUC	Operator code of grower	String	Yes	e.g. L1147
Orchard	Orchard code	String	No	e.g. JB1A
OrchardPhytoStatus	Status of orchard	String	Yes	e.g. EUA1-----
PHC	Packhouse code	String	Yes	e.g. L0748
ProductionArea	The region where produce derived	String	Yes	e.g. EC Standards = Province
Commodity	Commodity	String	Yes	e.g. SC Standards = Commodity

MarketingIndication	Market indicator	String	Yes	e.g. MRR Standards = MarketingIndication
FleshColour	Colour of product's flesh	String	No	e.g. White Flesh (Applies only to Peaches & Nectarines)
Class	Class	String	Yes	e.g. 1
ContainerNumber	Container's number	String	No	e.g. CRLU1319539
NumberOfPackagedItems	The number of packaged items on trade unit.	Int	Yes	e.g. 19
NetTradeUnitWeight	Trade Unit weight	Decimal	Yes	e.g. 304.234
NettPalletWeight	Pallet weight	Decimal	No	e.g. 304.234 (To be deprecated – use NetTradeUnitWeight)
PPECBInspectionDate	Date PPECB performed the inspection	String	Yes	W3 Format YYYY-MM-DD
StuffDate	Stuff date	String	Yes	W3 Format YYYY-MM-DD
InspectionManifestNo	Inspection Manifest No.	String	Yes	e.g. KO00010723
InspectionPointLocationCode	Inspection Point location code	String	Yes	e.g. 3055
<<<< TradeUnit				
<<< TradeUnits				
<< AdditionalInformation				
>> DisplayData (Properties used to override the information displayed on the Phyto document)				
NumberAndDescriptionOfPackage	Number and description of Package	String	No	e.g. 20 PALLETS 3600 CARTONS GRAPES
BotanicalName	The scientific name of the plant	String	No	e.g. CITRUS LIMON
DistinguishingMarks	Distinguishing markers	String	No	e.g. TRIU8999152
PlaceOfOrigin	Place of origin	String	No	e.g. SOUTH AFRICA – WESTERN CAPE

DeclaredMeansOfConveyance	Means of conveyance	String	No	e.g. MOL PROFICIENCY 196A
DeclaredPointOfEntry	First Point of Entry	String	No	e.g. Hamburg via Rotterdam
NameOfProductAndDeclaredQuantity	Name of the product and declared quantity	String	No	e.g. GRAPES - 16200.00 KG NETT
TreatmentDate	Date of treatment	String	No	e.g. 12 DECEMBER 2019 12H00
Treatment	Type of treatment	String	No	e.g. IN TRANSIT COLD TREATMENT
Chemical	Type of chemical applied	String	No	e.g. XXXXXXXXXXXXXXXX
DurationAndTemperature	Length of treatment and temperature applied	String	No	e.g. -0.55 C OR BELOW FOR 22 CONSECUTIVE DAYS
Concentration	The concentration of chemical application	String	No	e.g. XXXXXXXXXXXXXXXX
AdditionalInformation	Any additional treatment information	String	No	e.g. Date: 2011-11-30; Type:Chemical; Duration: 3 h; Temperature: 32 C; Concentration: 2.5 ml/kg; AdditionalInformation: No additional Information available
<<DisplayData				
>> CertificateFlexiFields				
>>> CertificateFlexiField				
Name	Name of the Flexi field	String	No	e.g. RegimeCode, ImportNo, etc.
Value	Value of Flexi field	Any	No	e.g. NZ738R
<<< CertificateFlexiField				
<< CertificateFlexiFields				
>> CertificateDocuments				
>>> CertificateDocument				

Name	The name of the document to be uploaded	String	Yes	e.g. Cold Treatment Certificate
DocumentTypeId	The type of document	Int	Yes	e.g. 1 Standards = DocumentType
MimeType	The mime type of the document	String	Yes	e.g. Standards = MimeType
Content	Serialized Document Data	String	Yes	Base64 Encoded String (i.e. Byte[] converted to Base64 code)
IsPrintable	Can the document be printed	Bool	No	Eg: false
<<< CertificateDocument				
<< CertificateDocuments				
< CertificateRequest				

Response:

Parameter	Description	Datatype	Example
ApplicationRefNo	The number generated when an application has been successfully created, used for reference purposes	String	e.g. 201900016
CustomReferenceNo	The consumer's number to reference back to their system (same as request)	String	e.g. 2016044
ApplicationStatus	The state of the application	String	e.g. New
DocumentsUploaded	The number of documents uploaded	Int	e.g. 3
DateCreated	The date the application was created	String	W3 Format YYYY-MM-DD
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

POST /api/v1/Certificate/Apply Application for a certificate to export

Response Class (Status 200)
Successfully submitted application

Model | Example Value

```
{
  "ApplicationRefNo": "string",
  "CustomReferenceNo": "string",
  "ApplicationStatus": "string",
  "DateCreated": "2020-04-15T19:20:36.131Z",
  "DocumentsUploaded": 0,
  "IsSuccessful": true,
  "Message": "string"
}
```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
request	(required)		body	Model Example Value
	Parameter content type: application/json			<pre>{ "ApplicationReferenceNo": "string", "ApplicationType": 0, "ApplicationStatus": "string", "CBRID": 0, "CBRBillingID": 0, "CustomReferenceNo": "string", "AgreementCode": "string", "DesiredIssueLocation": "string", "NotificationEmail": "string", "ConsignorId": "string", "ConsignorName": "string", }</pre>

1.1 Update ePhyto Certificate (eCert)-POST

To update a draft application, submit the same object as that for making an application as shown above in the **Apply for ePhyto Certificate** section.

Endpoint: refer to **Apply for ePhyto Certificate** section.

Request (xml, json): refer to **Apply for ePhyto Certificate** section.

Response: refer **Apply for ePhyto Certificate** section.

Please note:

1. On the initial application submission for each Consignment, the **SequenceNumeric** number is autogenerated by the system. To submit consignment items for the update, provide the **SequenceNumeric** number on each consignment so that the correct consignment is updated
2. To add any new child items on update e.g **Consignment, FlexiFields, TradeUnits** simply append the new object to the existing. To add new documents ,only send the new documents to be added.
3. To delete any child records e.g **FlexiFields, Consignments, TradeUnits**, omit them from the request when submitting for an update

Note: To simply add new documents to an existing Draft application without changing any other data, use the AddDocument endpoint. Please refer to the **Add Documents** section of this document for more information.

The request parameters to pay special attention to when submitting for an update are as follows:

ApplicationReferenceNo: Specify the application reference number to update

ApplicationStatus: “U” indicates update, if not specified a new application will be created

SequenceNumeric(Consignment Level): Uniquely identifies each consignment on an application

2. Get Certificate Data (eCert): GET

Used to retrieve the full details of an eCert application.

Endpoint:

- <https://app.ecert.co.za/api/v1/certificate/get> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/certificate/get> (TESTING)

Request (query parameter):

Parameter	Description	Datatype	Req?	Example/Ref
ApplicationRefNo	The identifier for Phyto application	String	Yes	e.g. 201900016

Response:

Parameter	Description	Datatype	Example/Ref
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. “Successfully submitted”
Data	Certificate data	Object	Please see ‘Apply ePhyto Request’

3. Get Certificate PDF (eCert): GET

Used to retrieve a pdf copy of the ePhyto certificate.

Endpoint:

- <https://app.ecert.co.za/api/v1/certificate/getpdf> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/certificate/getpdf> (TESTING)

Request (query parameter):

Parameter	Description	Datatype	Req?	Example/Ref
ApplicationRefNo	The identifier for Phyto application	String	Yes	e.g. 201900016

Response:

Parameter	Description	Datatype	Example/Ref
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"
Data	Certificate document	Byte[]	

4. Get Additional Input (eCert): GET

PLEASE NOTE: This feature is reserved for future purposes for cooling data.

Used to retrieve additional cooling data information for the application.

Endpoint:

- <https://app.ecert.co.za/api/v1/additionalinput> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/additionalinput> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/Ref
ApplicationRefNo	The identifier for Phyto application	String	Yes	e.g. 201900016

Response:

Parameter	Description	Datatype	Example/Ref
IsSuccessful	Response status	Boolean	e.g. true

ResponseMessage	Response message	String	e.g. "Successfully submitted"
AdditionalInput	Cooling data	Byte[]	e.g. pdf, excel, docx, documents

5. Post Additional Input (eCert): POST

PLEASE NOTE: This feature is reserved for future purposes for cooling data.

Used to post additional information for the certification process such as cooling data

Endpoint:

- <https://app.ecert.co.za/api/v1/additionalinput> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/additionalinput> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/Ref
ApplicationRefNo	The identifier for Phyto application	String	Yes	e.g. 201900016
AdditionalInput	Supporting documents	Byte[]	Yes	e.g. pdf, excel, docx, documents

Response:

Parameter	Description	Datatype	Example/Ref
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

6. Check Phyto Application Status (eCert): GET

Used to get the status of a phyto application after it has been submitted.

Endpoint:

- <https://app.ecert.co.za/api/v1/phytostatus/Get> (PRODUCTION)

- <http://qa.ecert.co.za/api/v1/phytostatus/Get> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/Ref
ApplicationRefNo	The identifier for Phyto application	String	Yes	e.g. 201900016

Response:

Parameter	Description	Datatype	Example/Ref
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"
HubTrackingNumber	The tracking number of the application on the IPPC Hub	String	e.g. ZANL54635645
HubTrackingStatus	A status message of the application	String	e.g. "Delivered"

Swagger Endpoint:

PhytoStatus
Show/Hide
List Operations
Expand Operations

GET /api/v1/PhytoStatus/Get
 Retrieve the status of an ePhyto application

Response Class (Status 200)
 OK

Model | Example Value

```
{}
```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
applicationRefNo	(required)		query	string

Try it out!

7. Add Documents (eCert):POST

Used to add documents to an existing draft application

Endpoint:

- <https://app.ecert.co.za/api/v1/Certificate/AddDocument> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/Certificate/AddDocument> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/ Ref
ApplicationReferenceNo	ApplicationReference Number to add the documents to	String	No	e.g. NPPO-ZA/2020/4/25001
>> CertificateDocuments				
>>> CertificateDocument				
Name	The name of the document to be uploaded	String	Yes	e.g. Cold Treatment Certificate
DocumentTypeeld	The type of document	Int	Yes	e.g. 1 Standards = DocumentType
MimeType	The mime type of the document	String	Yes	e.g. Standards = MimeType
Content	Serialized Document Data	String	Yes	Base64 Encoded String (i.e. Byte[] converted to Base64 code)
IsPrintable	Can the document be printed	Bool	No	Eg: false
<<< CertificateDocument				
<< CertificateDocuments				

Response:

Parameter	Description	Datatype	
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

POST /api/v1/Certificate/AddDocument Delete draft

Response Class (Status 200)
Document successfully uploaded

Model | **Example Value**

```
{
  "ApplicationRefNo": "string",
  "CustomReferenceNo": "string",
  "ApplicationStatus": "string",
  "DateCreated": "2020-08-07T13:29:54.967Z",
  "DocumentsUploaded": 0,
  "IsSuccessful": true,
  "Message": "string"
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
applicationReferenceNo	NPPO-ZA/2020/8/12345	Application Reference Number	query	string

certificateDocuments

```
[
  {
    "Name": "string",
    "MimeType": "string",
    "IsPrintable": true,
    "Content": "string",
    "DocumentTypeId": 0
  }
]
```

Parameter content type:

Lists of documents to be uploaded

body

Model | **Example Value**

```
[
  {
    "Name": "string",
    "MimeType": "string",
    "IsPrintable": true,
    "Content": "string",
    "DocumentTypeId": 0
  }
]
```

8. Get List of Agreements (eCert): GET

Used to get a list of available agreements

Endpoint:

- <https://app.ecert.co.za/api/v1/agreement/Get> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/agreement/Get> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/ Ref
CommodityCode	The commodityCode to filter by	String	No	e.g. GF- GRAPEFRUIT
CountryCode	The countryCode of a country to filter by	String	No	Netherlands - NL
exportDate	The date to filter on applicable agreements	Date-time	No	e.g 07/01/2020

Response:

Parameter	Description	Datatype	
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

GET
/v1/Agreement/Get
Retrieves list of Agreements

Response Class (Status 200)
Successfully retrieved list

Model | Example Value

```

{
  "restrictedValues": [
    {
      "key": "string",
      "value": "string"
    }
  ],
  "IsSuccessful": true,
  "Message": "string"
}

```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
commodityCode		The commodityCode to filter by e.g. GF - GRAPEFRUIT etc..	query	string
countryCode		The country code of a country to filter by e.g. Netherlands - NL	query	string
exportDate		The date to filter on agreements that are applicable (MM/dd/yyyy)	query	date-time

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Item invalid (code=400)	Model Example Value	
		<pre> { "IsSuccessful": true, "Message": "string" } </pre>	
404	No records found (code=404)	Model Example Value	
		<pre> { "IsSuccessful": true, "Message": "string" } </pre>	

Try it out!
Hide Response

9. Get List of Agreements by Code (eCert): GET

Used to get a list of available agreements by agreement code

Endpoint:

- <https://app.ecert.co.za/api/v1/Agreement/GetByAgreementCode> (PRODUCTION)
- <http://qa.ecert.co.za/API/v1/Agreement/GetByAgreementCode> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/ Ref
AgreementCode	The agreementCode to filter by	String	No	e.g Gernal_Burkina Faso – AGM0032

Response:

Parameter	Description	Datatype	
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

GET

/v1/Agreement/GetByAgreementCode

Get an agreement by agreement code

Response Class (Status 200)

Successfully an agreement by agreement code

Model

Example Value

```
{  "ModelListName": "string",  "Value": 0,  "RequirementType": 1,  "DisplayName": "string",  "AgreementId": 0},  {  "IsSuccessful": true,  "Message": "string"}
```

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
agreementCode	(required)		query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Item invalid (code=400)	Model Example Value <pre>{ "IsSuccessful": true, "Message": "string"}</pre>	

Try it out!

10. Get Data for Standard (eCert): GET

Used to get the list of standards by specifying the standard name

Endpoint:

- <https://app.ecert.co.za/api/v1/Standards/GetStandardData> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/Standards/GetStandardData> (TESTING)

Request (xml, json):

Parameter	Description	Datatype	Req?	Example/ Ref
StandardName	Name of the standard to retrieve	String	Yes	e.g PortOfEntry

Response:

Parameter	Description	Datatype	
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

GET /v1/Standards/GetStandardData Retrieves list of Standard List Data by specifying the name

Response Class (Status 200)
Successfully retrieved list

Model | **Example Value**

```
{
  "Data": [
    {}
  ],
  "IsSuccessful": true,
  "Message": "string"
}
```

Response Content Type (application/json)

Parameters

Parameter	Value	Description	Parameter Type	Data Type
standardName	(required)	Name of the Standard to retrieve	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Item invalid (code=400)	Model Example Value <pre>{ "IsSuccessful": true, "Message": "string" }</pre>	

[Try it out!](#)

11. List of Standards (eCert): Get

Used to retrieve a list of all available standards

Endpoint:

- <https://app.ecert.co.za/api/v1/Standards/GetStandardList> (PRODUCTION)
- <http://qa.ecert.co.za/api/v1/Standards/GetStandardList> (TESTING)

Response:

Parameter	Description	Datatype	
IsSuccessful	Response status	Boolean	e.g. true
ResponseMessage	Response message	String	e.g. "Successfully submitted"

Swagger Endpoint:

GET /v1/Standards/GetStandardList Retrieves list of available Standards

Response Class (Status 200)
Successfully retrieved list

Model | **Example Value**

```
[  
  "string"  
]
```

Response Content Type application/json ▼

Response Messages

HTTP Status Code	Reason	Response Model	Headers
404	Item not found (code=404)	Model Example Value <pre>{ "IsSuccessful": true, "Message": "string" }</pre>	

Try it out! [Hide Response](#)

API Change Log

Version 1.1 (09 September 2019)

- Changed Base API Url <https://www.ecert.co.za/api/v1> to <https://app.ecert.co.za/api/v1>
- Changed *Consignment.Common* to *Consignment.CommonName* (Apply for ePhyto Certificate)
- Changed *Addendum* to *AdditionalInformation* (Apply for ePhyto Certificate)
- Changed *Pallets* to *TradeUnits* (Apply for ePhyto Certificate)
- Changed *Pallet* to *TradeUnit* (Apply for ePhyto Certificate)
- Changed *NettPalletWeight* to *NettTradeUnitWeight* (Apply for ePhyto Certificate)
- Changed *NumberOfCartons* to *NumberOfPackagedItems* (Apply for ePhyto Certificate)
- *TradeUnit.FleshColour* changed to not required

Version 1.2 (30 September 2019)

- Changed fields *AppliedProcessStartDate*, *AppliedProcessEndDate*, *AppliedProcessTreatmentTypeLevel1* and *AppliedProcessTreatmentTypeLevel2* to optional fields
- Added *CertificateDocuments* to the API (Apply for ePhyto Certificate)
- *FlexiFields* now supported (Apply for ePhyto Certificate)
- Removed *RecipientId* (Apply for ePhyto Certificate)

Version 1.3 (14 October 2019)

- Added new endpoint to get certificate data (Get Certificate Data)
- Added new endpoint to get certificate copy of approved pdf (Get Certificate Document)
- Added *DurationMeasureCode* to *ConsignmentItem* (Apply for ePhyto)
- Added *DurationMeasure* to *ConsignmentItem* (Apply for ePhyto)
- Added *FullTreatmentInformation* to *ConsignmentItem* (Apply for ePhyto)
- Changed fields *ConsignorId* and *ConsigneeId* to not required (Apply for ePhyto Certificate)
- Changed datatypes for fields *NetWeightMeasure* and *GrossWeightMeasure* from *Int* to *Decimal* (Apply for ePhyto Certificate)
- Changed *CommodityVegetableClass* to *CommodityVegetablePart* (Apply for ePhyto Certificate)
- Changed *NetPalletWeight* to *NetTradeUnitWeight* (Apply for ePhyto Certificate)
- Changed datatype for field *NetTradeUnitWeight* from *Int* to *Decimal* (Apply for ePhyto Certificate)
- Removed *Description* from *ConsignmentItem* (Apply for ePhyto)
- Removed *AppliedProcessTypeCode* from *ConsignmentItem* (Apply for ePhyto)
- Removed *TargetCountry* from *AdditionalInformation* (Apply for ePhyto)
- Removed *TargetRegion* from *AdditionalInformation* (Apply for ePhyto)
- Removed *ExportCertificateNumber* from *AdditionalInformation* (Apply for ePhyto)
- Changed *ClientRef* to optional on *TradeUnit* (Apply for ePhyto)

- Changed *Orchard* to optional on TradeUnit for non-special markets (Apply for ePhyto)
- Changed *ProductionArea* to use 2-character code for areas e.g. Eastern Cape to EC
- Changed *Class* from *Int* to *String* on TradeUnit to cater for class 'P'
- Changed *ContainerNumber* to optional on TradeUnit (Apply for ePhyto)
- Removed *ScientificName* from ConsignmentItem (Apply for ePhyto)

Version 1.4 (21 January 2020)

- Added new Methods to get Agreements(Agreement/Get and GetByAgreementCode)
- Added new Methods to get Standards(GetStandardData and GetStandardList)
- Added *ScientificName* from ConsignmentItem (Apply for ePhyto)
- Added TargetCountry and TargetRegion (Apply for ePhyto)
- Added RegimeCode (Apply for ePhyto)

Version 1.4 (19 March 2020)

- Removed RegimeCode (Apply for ePhyto)

Version 1.4 (30 March 2020)

- Changed AppliedProcessStartDate - W3 Format YYYY-MM-DDThh:mmTZD (1997-07-16T19:20+01:00)
- Changed AppliedProcessEndDate - W3 Format YYYY-MM-DDThh:mmTZD (1997-07-16T19:20+01:00)

Version 1.4 (15 April 2020)

- Added *DocumentsUploaded* to the Response (Apply for ePhyto)
- Added *ApplicationReferenceNo* (Apply for ePhyto)
- Added *Description* to *ConsignmentItem* (Apply for ePhyto)
- Added *DisplayData* properties (Apply for ePhyto)
- Removed *AdditionalInformationID* from *TradeUnit* (Apply for ePhyto)

Version 1.4 (13 August 2020)

- Added *PaymentType* (Apply for ePhyto)
- Added *SequenceNumeric* (Apply for ePhyto)
- Added *How to Update Draft Application* section
- Added *Add Document* endpoint
- Removed Request Web API Access Application form (previously Appendix 1)

APPENDIX 1

Reference Documents

- How to Connect Web API Guide v 1.0
- CBR-Request Web API Access

Web References

- <https://oauthlib.readthedocs.io/en/latest/oauth2/grants/credentials.html>
- <https://oauth.net/2/grant-types/client-credentials/>