PSYCHE SYSTEMS

BACKBONE NUCLEOLIS API

REVISION 1.1
11/04/2020

TABLE OF CONTENTS

BACKBONE API

OVERVIEW – GETTING STARTED

This is a guide for using the Psyche Systems Backbone application programming interface (API).

This section of the guide covers getting started information for four essential requirements for communicating with the Backbone Server using the Backbone API:

1. Verifying/Confirming the Backbone Server Uniform Resource Locator (URL)
2. Logging in, authenticating, and establishing a session.
3. Calling API endpoints during a session.
4. Terminating a session and log out.

Appendix A provides more details about the available API commands that can be used for interacting with NucleoLIS.

Appendix B provides multiple coding examples illustrating how you can program a client side implementation to call the Backbone API.

THEORY OF OPERATION

The Backbone API is a Representational state transfer (REST) based API that is designed to run on Microsoft Internet Information Server (IIS). Typically, Psyche Systems deploys the server-side of the Backbone on IIS version 7 or later to host the Backbone Server API. The client side implementation of the API can be written in any programming language and run on any operating system as long as the client side implementation adheres to the standards outlined in this document.

To ensure Health Insurance Portability And Accountability Act (HIPAA) privacy compliance and to safeguard Protected Health Information (PHI), all requests to the Backbone Server API must use Hypertext Transfer Protocol Secure (HTTPS) over port 443.

The Backbone Server (being a REST based web server implementation) is considered to be stateless, with the exception of the "session" cookie which is passed back and forth between the client side implementation and the Backbone Server with each request after a successful logon is established.

SECURITY

To further ensure HIPAA compliance and security of PHI, the Backbone API provides session –based security as follows:

1. AUTHENTICATION.

A valid username and password is required to access the Backbone Server API. Once logged in, the API returns an access token that valid for 15 minutes and can be used to make subsequent requests. This access token is stored in a cookie which is transparently passed back and forth with each API request/response. This access token resets every time a successful API call is completed. With this approach, an application can authenticate once, make any number of API calls, and then log out, without needing to pass authentication credentials with every request.

2. SECURE TRANSPORT PROTOCOL (HTTP-S).

All API calls are made using Internet standard Hypertext Transport Protocol Secure (HTTPS) on port 443. The bidirectional encryption of communications between a client and server protects against eavesdropping and tampering of the communication. In practice, this provides a reasonable assurance that one is communicating without interference by attackers with the website that one intended to communicate with, as opposed to an impostor.

This implementation also uses the latest Transport Layer Security (TLS) encryption. This public-private key encryption overlays the symmetric AES encryption used for encrypting the payload, so essentially, the payload is double encrypted and is extremely secure.

3. SLIDING WINDOW TOKEN EXPIRY

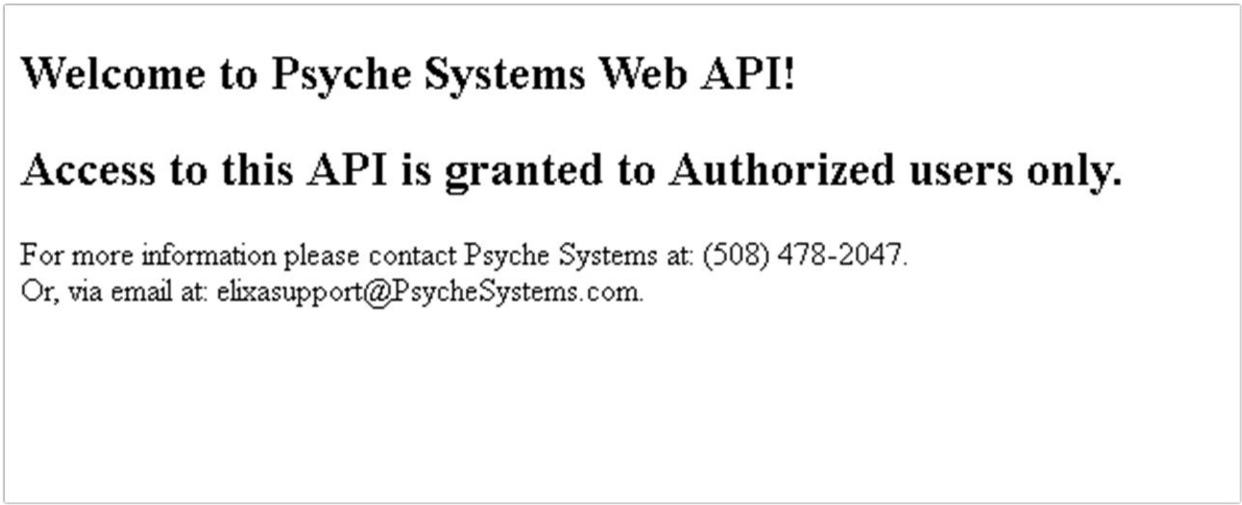
Each time a session token is provided to a requesting client, the token is encoded so that it is only valid for 15 minutes. This provides what is known as "sliding window" expiration. By using this security method, tokens that are copy/pasted can only be used for a limited period of time. Each time a successful request is made to the Backbone Server API, the token is refreshed to provide a new 15 minute window. In the event that no request is made within 15 minutes, the token expires and can no longer be used without logging in again to obtain a new token. This is the equivalent to the "idle timeout" feature that many online systems offer.

GETTING STARTED

For each of the examples shown, this document also provides in Appendix "B", coding examples in a wide variety of programming languages so the implementers can research options for creating their custom client implementation that communicates with the API.

STEP 1 – VERIFY/CONFIRM BACKBONE URL CONNECTIVITY

When first getting started with a client side implementation, it is helpful to first ensure that you can "reach" the Backbone Server from your client location. To do this, enter the URL of the Backbone Server that you were provided into any web browser. If you are able to connect using that URL, you will see a page that looks like this:



Welcome to Psyche Systems Web API!

Access to this API is granted to Authorized users only.

For more information please contact Psyche Systems at: (508) 478-2047.
Or, via email at: elixasupport@PsycheSystems.com.

If you don't see this page, please contact Psyche Systems support for assistance. If you do see this page, then this is the URL that you will use when making all API requests. This URL will be referred to as "https://www.SomeServer.com/api/" in the rest of this document.

STEP 2 – LOGGING INTO THE BACKBONE NUCLEOLIS API

Logging in, authenticates the requesting client as a valid user, and upon successful login, returns an authorization token that is stored in a session cookie. This cookie must be passed after each subsequent request.

This is accomplished by calling the "logon" endpoint with username and password information that is POSTED to the Backbone Server with the following settings:

URI	<API URL>/api/authenticate/logon
Host	Provide the hostname of the web server you are connecting with.
Authorization Method	No Auth
Content-Type	`application/x-www-form-urlencoded`
Content-Length	The length of the body payload in bytes. For example, if you are posting: `username=LAB&password=123456` Then, the Content-Length parameter would be set to 28. (Because there are 28 characters in the payload).
User-Agent	Set this to a unique name for your client application.
Accept	`*/*`
Posted Data	`username=<username>&password=<password>`

	Here, <username> and <password> are the username and password you were provided to access the API. If you do not have these credentials, please contact Psyche Systems support.

When the called API command succeeds, the response will look like this:

Status	200 OK
Body	<?xml version="1.0" encoding="UTF-8" ?><login>Success</login>
Set-Cookie	session=session-id=qAuOpF9tIBeBR8SIGLmDVRQGUFJ%2ftxLUbELwr8HtNxo%3d&username=346650&epoch=1604331210&timelimit=900; expires=Mon, 02 Nov 2020 15:48:30 GMT; domain=SomeServer.com; path=/; secure; httponly
Content-Type	Text/plain; charset=utf-8
Server	Microsoft-IIS/8.0

If the username or password is invalid, the response will look like this:

Status	401
Body	<?xml version="1.0" encoding="UTF-8" ?><login>username or password invalid</login
Set- Cookie	(No cookie returned)
Content-Type	Text/plain; charset=utf-8
Server	Microsoft-IIS/8.0

Other error messages can sometimes be shown as well. If you don't receive the word "Success" within the xml return message, then you should assume that the login failed, and the message returned is the error message.

Note: On initial login, there no cookie specified yet. The cookie gets created by the Backbone Server, and is returned to the client only after a successful logon request is executed. This "session" cookie will contain the authentication token and will look similar to this:

Cookie: `session=session-id=qAuOpF9tIBeBR8SIGLmDVRQGUFJ%2ftxLUbELwr8HtNxo%3d&username=346650&e`

```
poch=1604331210&timelimit=900; expires=Mon, 02 Nov 2020 15:48:30 GMT;
domain=SomeServer.com; path=/; secure; httponly
```

This cookie will be updated after each successful request, and must be passed back to the server on each subsequent request. The session token contained within the cookie is only valid for 15 minutes. If a subsequent API endpoint request is not made within the session "timeout" limit of 15 minutes, then the session token becomes invalid, and the client will need to login again to get a new session token by calling the "logon" API endpoint again.

See Appendix "B" for programming examples that call this endpoint.

STEP 3- CALLING API ENDPOINTS WITHIN A SESSION

Once you are logged in, the next step is to call an API Endpoint using GET. In this example we will show how to call the NucleoLIS api/N/GetHeartbeat API endpoint. This API endpoint returns a heartbeat message so you can determine if the API is able to communicate with NucleoLIS.

URI	<API URL>/ api/N/GetHeartbeat
Host	Provide the hostname of the web server you are connecting with.
Authorization Method	No Auth
Cookie	Pass in the cookie you received in the response from the logon API call. It should look similar to this: `session=session-id=H38%2fAImLMVMNssLZsY%2fx7mWkfNsHIM4m1CVO%2bFDc9aU%3d&username=346650&epoch=1604330631&timelimit=900`
User-Agent	Set this to a unique name for your client application.
Accept	`*/*`

When the called API command succeeds, the response will look like this:

Status	`200 OK`
Body	`<?xml version="1.0" encoding="UTF-8" ?><Result>OK</Result>`
Set-Cookie	`session=session-id=xjxAnqMxbu9D%2fbYakV5nRMeKxLlEVnADTiMJqUJycp0%3d&username=346650&epoch=1604330884&timelimit=900; expires=Mon, 02 Nov 2020 15:43:04 GMT; domain=SomeServer.com; path=/; secure; httponly`
Content-Type	`Text/plain; charset=utf-8`
Server	`Microsoft-IIS/8.0`

See Appendix “B” for programming examples that call this endpoint.

STEP 4- TERMINATE A SESSION AND LOG OUT

When you are done communicating with the Backbone Server API, there are two choices on how to end the session.

1. Do nothing – the cookie containing the latest session token will expire after 15 minutes.
2. Call the Logout API endpoint to destroy the cookie and immediately terminate the session:

URI	<API URL>/ api/authenticate/logout
Host	Provide the hostname of the web server you are connecting with.
Authorization Method	No Auth
Cookie	Pass in the cookie you received in the response from the logon API call. It should look similar to this: `session=session-id=H38%2fAImLMVMNssLZsY%2fx7mWkfNsHIM4m1CVO%2bFDc9aU%3d&username=346650&epoch=1604330631&timelimit=900`
User-Agent	Set this to a unique name for your client application.
Accept	`*/*`

When the called API command succeeds, the response will look like this:

Status	`200 OK`
Body	`<?xml version=”1.0” encoding=”UTF-8” ?><logout>Success</logout>`
Cookie	`(The cookie no longer exists)`
Content-Type	`Text/plain; charset=utf-8`

APPENDIX A – NUCLEOLIS API ENDPOINTS

INTRODUCTION

In addition to the logon and logoff API endpoints covered in section 1, the endpoints in this appendix are all of the available API endpoints for NucleoLIS Interaction.

GET HEARTBEAT

Endpoint	api/N/GetHeartbeat
Purpose	Returns a NucleoLIS heartbeat message so you can determine if the API is able to communicate with NucleoLIS system.
Request Type	GET
Parameters	{None}
Example	`https://www.SomeServer.com/api/N/GetHeartbeat`
Returns	If the heartbeat succeeds, returns: `<?xml version=”1.0” encoding=”UTF8” ?><Result>OK</Result>`

GET PATIENT

Endpoint	api/N/GetPatient
Purpose	Returns a single, full NucleoLIS patient record based on a patient unique identifier.
Request Type	GET

Parameters	`patient_id={patient_id}`
	`user_id={user_id}`

	return_format={return_format} Where: **patient_id** is the objectID of the requested Patient record. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/GetPatient?patient_id=518535440&user_id=245654&return_format=0

Returns	If the request succeeds, returns a patient record in the following XML format:
	<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
	  <PatientTable>
	    <Patient_ObjectID>518535440</Patient_ObjectID>
	    <Patient_CreationDate>2020-11-02T11:23:31.247-05:00</Patient_CreationDate>
	    <Patient_UpdateTime>2020-11-02T11:23:31.247-05:00</Patient_UpdateTime>
	    <Patient_Name>Boop, Betty</Patient_Name>
	    <Patient_Sex>F</Patient_Sex>
	    <Patient_Street />
	    <Patient_City />
	    <Patient_State />
	    <Patient_Zip />
	    <Patient_Phone />
	    <Patient_Fax />
	    <Patient_Code>518535440</Patient_Code>
	    <Patient_Soundex>BOPBTY</Patient_Soundex>
	    <Patient_Dob>1965-03-23</Patient_Dob>
	    <Patient_Ssn>011-22-3333</Patient_Ssn>
	    <Patient_MedicareNumber />
	    <Patient_Status>I</Patient_Status>
	    <Patient_CREATOR>LAB</Patient_CREATOR>
	    <Patient_RACE>White</Patient_RACE>
	    <Patient_STREET2 />
	    <Patient_USER1 />
	    <Patient_USER2 />
	    <Patient_USER3 />
	    <Patient_USER4 />
	    <Patient_MODIFIER>LAB</Patient_MODIFIER>
	    <Patient_LastName>Boop</Patient_LastName>

	<Patient_FirstName>Betty</Patient_FirstName> <Patient_MiddleName /> <Case_Status>I</Case_Status> </PatientTable> </NewDataSet>

GET PATIENTS

Endpoint	api/N/GetPatients
Purpose	Returns a collection of NucleoLIS patient records based on filter parameters passed.
Request Type	GET

Parameters	last_name = {last_name} (use * for wildcard)
	first_name = {first_name} (use * for wildcard)
	activesOnly = {activesOnly} (true, or false)
	filter_expression = {filter_expression}
	user_id = {user_id}
	return_format = {return_format}
	
	Where:
	last_name is the last name of the patient(s) to search for. Wildcards are supported. (Example S* will return all patients with a last name beginning with ‘S’.)
	first_name is the first name of the patient(s) to search for. Wildcards are supported. (Example L* will return all patients with a first name beginning with ‘L’.)
	activesOnly is a true or false parameter. If true, only active patients will be returned. If false, all patients (both active and inactive) will be returned.
	filter_expression is used as a secondary custom filtering criteria applied to the result set . Wildcards are supported. It can be used with any column returned in the result set. Any valid SQL WHERE clause is accepted.
	Multiple columns can be used when separated with ‘AND’. Example: Patient_State=’MA’ AND Patient_City=’W*’ will return all patients that live in MA in a city that begins with ‘W’.

user_id is the user code for the user in the NucleoLIS user table that is requesting the record.

return_format is the format for returning the data. Where:

```
    0 = XML Format

    1 = JSON Format
```

Example

```
https://www.SomeServer.com/api/N/GetPatients?last_name=S
*&first_name=*&activesOnly=true&filter_expression=Patien
t_Sex='Male'&user_id=245654&return_format=0
```

Returns

If the request succeeds, returns a collection of 1..n patient records matching the filtering criteria in the following XML format:

```
<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
  <PatientTable>
    <Patient_ObjectID>433016111</Patient_ObjectID>
    <Patient_Name>Claus, Santa</Patient_Name>
    <Patient_Code>433016111</Patient_Code>
    <Patient_Street />
    <Patient_STREET2 />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Sex>Male</Patient_Sex>
    <Patient_Dob>1900-12-25</Patient_Dob>
    <Patient_Ssn />
  </PatientTable>
  <PatientTable>
    <Patient_ObjectID>503609702</Patient_ObjectID>
    <Patient_Name>Claus, Santa</Patient_Name>
    <Patient_Code>503609702</Patient_Code>
    <Patient_Street />
    <Patient_STREET2 />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Sex>Male</Patient_Sex>
    <Patient_Dob>1900-12-25</Patient_Dob>
    <Patient_Ssn />
  </PatientTable>
  <PatientTable>
    <Patient_ObjectID>504251809</Patient_ObjectID>
    <Patient_Name>Claus, Santa</Patient_Name>
    <Patient_Code>504251809</Patient_Code>
    <Patient_Street />
    <Patient_STREET2 />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Sex>Male</Patient_Sex>
    <Patient_Dob>1900-12-25</Patient_Dob>
    <Patient_Ssn />
  </PatientTable>
  <PatientTable>
    <Patient_ObjectID>4550996</Patient_ObjectID>
    <Patient_Name>Sanding, Sammie</Patient_Name>
    <Patient_Code>4550996</Patient_Code>
    <Patient_Street />
    <Patient_STREET2 />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Sex>Male</Patient_Sex>
    <Patient_Dob>1980-04-22</Patient_Dob>
    <Patient_Ssn />
  </PatientTable>
  <PatientTable>
    <Patient_ObjectID>507538376</Patient_ObjectID>
    <Patient_Name>Sanding, Sammie</Patient_Name>
    <Patient_Code>507538376</Patient_Code>
    <Patient_Street />
    <Patient_STREET2 />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Sex>Male</Patient_Sex>
    <Patient_Dob>1980-04-22</Patient_Dob>
    <Patient_Ssn />
  </PatientTable>

</NewDataSet>
```

GET CASE

Endpoint	api/N/GetCase
Purpose	Return a single, full NucleoLIS case record based on case number. The full case record inclusive of all data elements associated with patient, case, specimen, test order and test result records are returned.
Request Type	GET

Parameters

```
case_number={case_number}
user_id={user_id}
return_format={return_format}
```

Where:

case_number is the number of the case.

user_id is the user code for the user in the NucleoLIS user table that is requesting the record.

return_format is the format for returning the data. Where:

```
    0 = XML Format
    1 = JSON Format
```

Example

```
https:// www.SomeServer.com /api/N/GetCase?case_number=M2020-000032&user_id=245654&return_format=0
```

Returns

If the request succeeds, returns the requested case record in the following XML format:

```
<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
  <CaseTable>
    <Patient_ObjectID>163307929</Patient_ObjectID>
    <Patient_CreationDate>2016-11-10T16:11:29.84-05:00</Patient_CreationDate>
    <Patient_UpdateTime>2019-05-22T10:51:15.137-04:00</Patient_UpdateTime>
    <Patient_Name>O'Hara, Jaime</Patient_Name>
    <Patient_Sex>Female</Patient_Sex>
    <Patient_Street />
    <Patient_City />
    <Patient_State />
    <Patient_Zip />
    <Patient_Phone />
    <Patient_Fax />
    <Patient_Code>163307929</Patient_Code>
    <Patient_Soundex>O'HRJM</Patient_Soundex>
    <Patient_Dob>2018-05-16</Patient_Dob>
    <Patient_Ssn />
    <Patient_MedicareNumber />
    <Patient_Status>I</Patient_Status>
    <Patient_CREATOR>LAB</Patient_CREATOR>
    <Patient_RACE />
    <Patient_STREET2 />
    <Patient_USER1 />
    <Patient_USER2 />
    <Patient_USER3 />
    <Patient_USER4 />
    <Patient_Modifier>Scaron</Patient_Modifier>
    <Patient_LastName>O'Hara</Patient_LastName>
    <Patient_FirstName>Jaime</Patient_FirstName>
    <Visit_ObjectID>498604164</Visit_ObjectID>
    <Visit_CreationDate>2019-05-22T10:51:34.293-
04:00</Visit_CreationDate>
    <Visit_UpdateTime>2019-05-22T10:51:34.293-
04:00</Visit_UpdateTime>

<Visit_SuperobjectID>163307929</Visit_SuperobjectID>
    <Visit_Patient>163307929</Visit_Patient>
    <Visit_Code>498604164</Visit_Code>
    <Visit_Location />
    <Visit_Type>OP</Visit_Type>
    <Visit_VisitDate>2019-05-22</Visit_VisitDate>
    <Visit_Status>I</Visit_Status>
    <Visit_DischargeDate />
    <Visit_ExtraField1 />
    <Visit_ExtraField2 />
    <Visit_CREATOR>Scaron</Visit_CREATOR>
    <Visit_USER1 />
    <Visit_USER2 />
    <Visit_USER3 />
    <Visit_USER4 />
    <Visit_comment />
    <Case_ObjectID>506873424</Case_ObjectID>
    <Case_CreationDate>2020-03-09T14:50:06.023-
04:00</Case_CreationDate>
    <Case_UpdateTime>2020-05-11T14:03:15.34-
04:00</Case_UpdateTime>
    <Case_SuperobjectID>498604164</Case_SuperobjectID>
    <Case_User1 />
    <Case_User2 />
    <Case_User3 />
    <Case_User4 />
    <Case_CREATOR>Scaron</Case_CREATOR>
    <Case_MODIFIER>Bishop</Case_MODIFIER>
    <Case_Number>M2020-000032</Case_Number>
    <Case_ReqNumber />
    <Case_ReportedFlag>False</Case_ReportedFlag>

<Case_SubmitterLocation>258012099</Case_SubmitterLocat
ion>
    <Case_OrderID />
    <Case_Status>I</Case_Status>
    <Case_ReportStatus>FINAL</Case_ReportStatus>
    <Case_CaseType>14662557</Case_CaseType>
    <Case_CorrectedComment />
    <Case_AmendedComment />
    <Case_ReportedBy>Bishop</Case_ReportedBy>
    <Case_CanceledBy />
    <Case_CanceledDate />
    <Case_ReviewFlag>2020-03-09</Case_ReviewFlag>
    <Case_ReviewedBy />
    <Case_ReviewedDate />
    <Case_ReviewedComment />
    <Case_ReviewedTime />
    <Case_StatusStep />
    <Case_CanceledFor />
    <Case_Signer>370583063</Case_Signer>
    <Case_SignerDate>2020-03-09</Case_SignerDate>
    <Case_ICDs />
    <Case_ClinicalInformation />
    <Case_Images><?xml version="1.0" encoding="utf-
16"?>
<ChrysalisImgDats
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
/></Case_Images>
    <Case_DocumentID />
    <Case_OrderDate>2020-03-09</Case_OrderDate>
    <Case_OrderTime />
    <Case_ExternalID />
    <Case_ExternalPatientID />
    <Case_ReopenReason />
    <Case_ReopenDate />
    <Case_Signatures><?xml version="1.0"
encoding="utf-16"?>
<CaseSignatureHistory
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Sigs>
    <CaseSignature>
      <Signer>370583063</Signer>
      <SignerDate>2020-03-09T15:08:36</SignerDate>
      <ReOpenReason />
      <ReOpener>0</ReOpener>
      <ReOpenedDate>0001-01-01T00:00:00</ReOpenedDate>
      <UnSign>false</UnSign>
    </CaseSignature>
    <CaseSignature>
      <Signer>370583063</Signer>
      <SignerDate>2020-03-09T15:10:42</SignerDate>
      <ReOpenReason />
      <ReOpener>0</ReOpener>
      <ReOpenedDate>0001-01-01T00:00:00</ReOpenedDate>
      <UnSign>true</UnSign>
    </CaseSignature>
    <CaseSignature>
      <Signer>370583063</Signer>
      <SignerDate>2020-03-09T15:11:09</SignerDate>
      <ReOpenReason />
      <ReOpener>0</ReOpener>
      <ReOpenedDate>0001-01-01T00:00:00</ReOpenedDate>
      <UnSign>false</UnSign>
    </CaseSignature>
  </Sigs>
</CaseSignatureHistory></Case_Signatures>
    <Case_NeedsCorrectedF>False</Case_NeedsCorrectedF>
    <Case_PrescriptionDrugs><?xml version="1.0"
encoding="utf-16"?>
<PrescribedDrugs
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
/></Case_PrescriptionDrugs>
    <Case_DataCollect />
    <Case_ReportIX>1</Case_ReportIX>
    <Case_OInitStatus>I</Case_OInitStatus>

<Case_RequireSignature>True</Case_RequireSignature>
    <Case_SignerTime>15:11</Case_SignerTime>
    <Case_CorrectionRequest />
    <Case_CorrectionRequestDate />
    <Case_CorrectionRequestTime />

<Case_SignerLocation>246584261</Case_SignerLocation>
    <Case_ClearCorrectedReason />
    <Case_QuestionsStash><?xml version="1.0"
encoding="utf-16"?>
<AAOEStasherBase
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Responses>
    <AAOEItemBase>
      <QObjectID>302726734</QObjectID>
      <Question>Additional Information</Question>
      <AllowOther>true</AllowOther>
      <SingleSelect>false</SingleSelect>
      <Required>true</Required>
      <FullResponse />
      <GridQuestion>false</GridQuestion>
      <GridQuestions />
      <Answers>
        <string>none</string>
      </Answers>
    </AAOEItemBase>
    <AAOEItemBase>
      <QObjectID>481211124</QObjectID>
      <Question>General Health</Question>
      <AllowOther>true</AllowOther>
      <SingleSelect>true</SingleSelect>
      <Required>false</Required>
      <FullResponse />
      <GridQuestion>false</GridQuestion>
      <GridQuestions />
      <Answers>
        <string>Good</string>
      </Answers>
    </AAOEItemBase>
    <AAOEItemBase>
      <QObjectID>481215687</QObjectID>
      <Question>Fasting</Question>
      <AllowOther>false</AllowOther>
      <SingleSelect>true</SingleSelect>
      <Required>true</Required>
      <FullResponse />
      <GridQuestion>false</GridQuestion>
      <GridQuestions />
      <Answers>
        <string>Yes</string>
      </Answers>
    </AAOEItemBase>
  </Responses>
</AAOEStasherBase></Case_QuestionsStash>
    <Case_Questions>Additional Information:none
General Health:Good
Fasting:Yes</Case_Questions>

<Case_QuestionsComplete>True</Case_QuestionsComplete>
    <Specimen_ObjectID>506873435</Specimen_ObjectID>
    <Specimen_CreationDate>2020-03-09T14:50:41.513-
04:00</Specimen_CreationDate>
    <Specimen_UpdateTime>2020-03-09T14:51:10.153-
04:00</Specimen_UpdateTime>

<Specimen_SuperobjectID>506873424</Specimen_Superobjec
tID>
    <Specimen_User1 />
    <Specimen_User2 />
    <Specimen_User3 />
    <Specimen_User4 />
    <Specimen_CREATOR>Scaron</Specimen_CREATOR>
    <Specimen_MODIFIER>Scaron</Specimen_MODIFIER>
    <Specimen_CollectionDate>2020-03-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>09:00</Specimen_CollectionTim
e>
```

```
    <Specimen_ReceivedDate>2020-03-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>14:51:10</Specimen_ReceivedTime
>
    <Specimen_Receiver>Scaron</Specimen_Receiver>
    <Specimen_Number>506873435</Specimen_Number>
    <Specimen_Status>A</Specimen_Status>
    <Specimen_Comment />
    <Specimen_Description />

<Specimen_SourceDefinition>4348467</Specimen_SourceDef
inition>
    <Specimen_Interpretation />
    <Specimen_StatusStep />
    <Specimen_SuperOrders />
    <Specimen_UnInlabDate />
    <Specimen_DataCollect />
    <Specimen_Priority>Routine</Specimen_Priority>
    <Specimen_Site />
    <Specimen_Gross />
    <Specimen_AddOnDate />
    <Specimen_AddOnTime />
    <Specimen_AddOn>False</Specimen_AddOn>
    <Specimen_MultipleCountID />
    <Specimen_MergeComments />
    <Specimen_MergeIntoComments />
    <Specimen_QuestionsStash />
    <Specimen_Questions />

<Specimen_QuestionsComplete>True</Specimen_QuestionsCo
mplete>
    <Specimen_FastingStatus />
    <Profile_TYPE>IMMUNO</Profile_TYPE>
    <Profile_ObjectID>506873438</Profile_ObjectID>
    <Profile_CreationDate>2020-03-09T14:50:49.687-
04:00</Profile_CreationDate>
    <Profile_UpdateTime>2020-03-09T15:07:50.84-
04:00</Profile_UpdateTime>

<Profile_SuperobjectID>506873435</Profile_SuperobjectI
D>
    <Profile_User1 />
    <Profile_User2 />
    <Profile_User3 />
    <Profile_User4 />
    <Profile_CREATOR>Scaron</Profile_CREATOR>
    <Profile_MODIFIER>Suec</Profile_MODIFIER>
    <Profile_Description />
    <Profile_OrderDate>2020-03-09</Profile_OrderDate>
    <Profile_OrderTime>14:50</Profile_OrderTime>
    <Profile_OrderedBy>Scaron</Profile_OrderedBy>

<Profile_PanelDefinition>416976655</Profile_PanelDefin
ition>
    <Profile_R1 />
    <Profile_R2 />
    <Profile_R3 />
    <Profile_R4 />
    <Profile_R5 />
    <Profile_R6 />
    <Profile_Result />
    <Profile_Reportable>False</Profile_Reportable>

<Profile_DatumEntryDefinition>478882298</Profile_Datum
EntryDefinition>
    <Profile_Images><?xml version="1.0" encoding="utf-
16"?>
<ChrysalisImgDats
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
/></Profile_Images>
    <Profile_Abnormal>False</Profile_Abnormal>
    <Profile_AOV>False</Profile_AOV>
    <Profile_StatusStep />
    <Profile_Range />
    <Profile_RangeComment />
    <Profile_CanceledDate />
    <Profile_CanceledBy />
    <Profile_CanceledFor />
    <Profile_ReflexedBy />
    <Profile_Signer>270861542</Profile_Signer>
    <Profile_SignerDate>2020-03-
09</Profile_SignerDate>
    <Profile_IOrder />
    <Profile_IOrderDate />
    <Profile_IOrderTime />
    <Profile_Complete>True</Profile_Complete>
    <Profile_DataCollect />
    <Profile_Signatures><?xml version="1.0"
encoding="utf-16"?>
<ProfileSignatureHistory
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Sigs>
    <ProfileSignature>
      <Signer>270861542</Signer>
      <SignerDate>2020-03-09T15:07:50</SignerDate>
      <UnSign>false</UnSign>
    </ProfileSignature>
  </Sigs>
</ProfileSignatureHistory></Profile_Signatures>
```

```
<Profile_RequireSignature>True</Profile_RequireSignatu
re>
    <Profile_Approver>162355184</Profile_Approver>
    <Profile_ApproveDate>2020-03-
09</Profile_ApproveDate>
    <Profile_Approvals><?xml version="1.0"
encoding="utf-16"?>
<ProfileApprovalHistory
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Sigs>
    <ProfileApproval>
      <Signer>162355184</Signer>
      <SignerDate>2020-03-09T14:56:31</SignerDate>
      <UnSign>false</UnSign>
    </ProfileApproval>
  </Sigs>
</ProfileApprovalHistory></Profile_Approvals>

<Profile_RequireApproval>True</Profile_RequireApproval
>
    <Profile_SignerTime>15:07</Profile_SignerTime>
    <Profile_ApproveTime>14:56</Profile_ApproveTime>
    <Profile_WPDLink />
    <Profile_CompleteDate />
    <Profile_CompleteTime />

<Profile_SignerLocation>246584261</Profile_SignerLocat
ion>

<Profile_PendingReagent>False</Profile_PendingReagent>
    <Profile_ORDLink />
    <Profile_QuestionsStash><?xml version="1.0"
encoding="utf-16"?>
<AAOEStasherBase
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
/></Profile_QuestionsStash>
    <Profile_Questions />

<Profile_QuestionsComplete>True</Profile_QuestionsComp
lete>
    <Profile_ROrder>I</Profile_ROrder>
    <Profile_ROrderDate>2020-03-
09</Profile_ROrderDate>
    <Profile_ROrderTime>14:50</Profile_ROrderTime>
    <Profile_ROrderLab>111200992</Profile_ROrderLab>
    <Profile_Interpretation />
```

```
<Profile_ProfileDefinitionObjectID>416976655</Profile_ProfileDefinitionObjectID>
    <Profile_ProfileDefinitionCreationDate>2018-10-31T10:16:34.31-04:00</Profile_ProfileDefinitionCreationDate>
    <Profile_ProfileDefinitionUpdateTime>2020-03-09T14:48:48.757-04:00</Profile_ProfileDefinitionUpdateTime>
    <Profile_ProfileDefinitionUser1 />
    <Profile_ProfileDefinitionUser2 />
    <Profile_ProfileDefinitionUser3 />
    <Profile_ProfileDefinitionUser4 />

<Profile_ProfileDefinitionCREATOR>debbieb</Profile_ProfileDefinitionCREATOR>

<Profile_ProfileDefinitionMODIFIER>Scaron</Profile_ProfileDefinitionMODIFIER>

<Profile_ProfileDefinitionCode>AMPH</Profile_ProfileDefinitionCode>

<Profile_ProfileDefinitionDescription>Amphetamine</Profile_ProfileDefinitionDescription>

<Profile_ProfileDefinitionRETIRED>False</Profile_ProfileDefinitionRETIRED>

<Profile_ProfileDefinitionDatumEntryDefinition>478882298</Profile_ProfileDefinitionDatumEntryDefinition>
    <Profile_ProfileDefinitionAlwaysReportMessage />
    <Profile_ProfileDefinitionMethodology />

<Profile_ProfileDefinitionRequireSignature>True</Profile_ProfileDefinitionRequireSignature>
    <Profile_ProfileDefinitionCodeMap />

<Profile_ProfileDefinitionLabelCount>1</Profile_ProfileDefinitionLabelCount>

<Profile_ProfileDefinitionAutoComplete>False</Profile_ProfileDefinitionAutoComplete>
    <Profile_ProfileDefinitionConstituentPanels><?xml version="1.0" encoding="utf-16"?>
<GPathConstituentPanels xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" /></Profile_ProfileDefinitionConstituentPanels>
```

```
<Profile_ProfileDefinitionAllowDuplicate>False</Profile_ProfileDefinitionAllowDuplicate>
    <Profile_ProfileDefinitionPendingMessage />

<Profile_ProfileDefinitionWPLink>False</Profile_ProfileDefinitionWPLink>

<Profile_ProfileDefinitionWPDiscrete>False</Profile_ProfileDefinitionWPDiscrete>

<Profile_ProfileDefinitionWPCyto>False</Profile_ProfileDefinitionWPCyto>

<Profile_ProfileDefinitionApprovalSignature>True</Profile_ProfileDefinitionApprovalSignature>

<Profile_ProfileDefinitionLabelDefinition>318256</Profile_ProfileDefinitionLabelDefinition>

<Profile_ProfileDefinitionWPDiscreteAdvanceOnly>False</Profile_ProfileDefinitionWPDiscreteAdvanceOnly>

<Profile_ProfileDefinitionRequireLocation>False</Profile_ProfileDefinitionRequireLocation>

<Profile_ProfileDefinitionUseDataCSort>False</Profile_ProfileDefinitionUseDataCSort>

<Profile_ProfileDefinitionAllowBatchResult>True</Profile_ProfileDefinitionAllowBatchResult>

<Profile_ProfileDefinitionIsToxicology>True</Profile_ProfileDefinitionIsToxicology>

<Profile_ProfileDefinitionIsClinical>False</Profile_ProfileDefinitionIsClinical>

<Profile_ProfileDefinitionSuppressOverlap>False</Profile_ProfileDefinitionSuppressOverlap>

<Profile_ProfileDefinitionOutreachCode>AMPH</Profile_ProfileDefinitionOutreachCode>

<Profile_ProfileDefinitionOutreachDescription>Amphetamine</Profile_ProfileDefinitionOutreachDescription>

<Profile_ProfileDefinitionAllowReferenceLab>True</Profile_ProfileDefinitionAllowReferenceLab>
```

```
<SourceDefinition_ObjectID>4348467</SourceDefinition_O
bjectID>
    <SourceDefinition_CreationDate>2015-04-
23T09:31:01.957-04:00</SourceDefinition_CreationDate>
    <SourceDefinition_UpdateTime>2020-09-
18T16:59:13.113-04:00</SourceDefinition_UpdateTime>
    <SourceDefinition_User1 />
    <SourceDefinition_User2 />
    <SourceDefinition_User3 />
    <SourceDefinition_User4 />

<SourceDefinition_CREATOR>deannac</SourceDefinition_CR
EATOR>

<SourceDefinition_MODIFIER>Scaron</SourceDefinition_MO
DIFIER>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition_
Description>

<SourceDefinition_RETIRED>False</SourceDefinition_RETI
RED>

<SourceDefinition_LabelCount>1</SourceDefinition_Label
Count>

<SourceDefinition_WPLink>False</SourceDefinition_WPLin
k>

<SourceDefinition_WPRequireProfiles>False</SourceDefin
ition_WPRequireProfiles>

<SourceDefinition_WPAllowNOrdering>False</SourceDefini
tion_WPAllowNOrdering>

<SourceDefinition_ForeignAllowNOrdering>False</SourceD
efinition_ForeignAllowNOrdering>

<SourceDefinition_WPAddDiagnosis>False</SourceDefiniti
on_WPAddDiagnosis>

<SourceDefinition_LabelDefinition>280272780</SourceDef
inition_LabelDefinition>
    <SourceDefinition_Sites><?xml version="1.0"
encoding="utf-16"?>
<XStrings xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
/></SourceDefinition_Sites>
```

```
<SourceDefinition_OverdueDefault>8</SourceDefinition_OverdueDefault>

<SourceDefinition_Multiples>2</SourceDefinition_Multiples>

<SourceDefinition_AutoMultiples>False</SourceDefinition_AutoMultiples>

<SourceDefinition_GrossReceived>False</SourceDefinition_GrossReceived>

<SourceDefinition_AAOESet>485355213</SourceDefinition_AAOESet>

<SourceDefinition_IgnoreLabelDefault>False</SourceDefinition_IgnoreLabelDefault>

<CaseTypeDefinition_ObjectID>14662557</CaseTypeDefinition_ObjectID>
    <CaseTypeDefinition_CreationDate>2015-06-25T11:10:41.087-04:00</CaseTypeDefinition_CreationDate>
    <CaseTypeDefinition_UpdateTime>2020-10-27T11:57:35.247-04:00</CaseTypeDefinition_UpdateTime>
    <CaseTypeDefinition_User1 />
    <CaseTypeDefinition_User2 />
    <CaseTypeDefinition_User3 />
    <CaseTypeDefinition_User4 />

<CaseTypeDefinition_CREATOR>0</CaseTypeDefinition_CREATOR>

<CaseTypeDefinition_MODIFIER>Scaron</CaseTypeDefinition_MODIFIER>

<CaseTypeDefinition_Code>MOLC</CaseTypeDefinition_Code>

<CaseTypeDefinition_Description>Molecular</CaseTypeDefinition_Description>

<CaseTypeDefinition_RETIRED>False</CaseTypeDefinition_RETIRED>

<CaseTypeDefinition_ReportDefinition>217397414</CaseTypeDefinition_ReportDefinition>

<CaseTypeDefinition_OverdueDays>1</CaseTypeDefinition_OverdueDays>
```

```
<CaseTypeDefinition_RequirePreliminary>False</CaseTypeDefinition_RequirePreliminary>

<CaseTypeDefinition_StepDefinition>245594</CaseTypeDefinition_StepDefinition>

<CaseTypeDefinition_RequireSignature>False</CaseTypeDefinition_RequireSignature>

<CaseTypeDefinition_AccessionLetter>M</CaseTypeDefinition_AccessionLetter>

<CaseTypeDefinition_UseProfileReports>False</CaseTypeDefinition_UseProfileReports>

<CaseTypeDefinition_PrelimReportDefinition>217397414</CaseTypeDefinition_PrelimReportDefinition>

<CaseTypeDefinition_WPLink>False</CaseTypeDefinition_WPLink>

<CaseTypeDefinition_WPSignOnReport>False</CaseTypeDefinition_WPSignOnReport>

<CaseTypeDefinition_WPLoadReport>False</CaseTypeDefinition_WPLoadReport>

<CaseTypeDefinition_WPAllowSupplemental>False</CaseTypeDefinition_WPAllowSupplemental>

<CaseTypeDefinition_LabelDefinition>348382382</CaseTypeDefinition_LabelDefinition>
    <CaseTypeDefinition_WPSignSuppress><?xml version="1.0" encoding="utf-16"?>
<XStrings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" /></CaseTypeDefinition_WPSignSuppress>

<CaseTypeDefinition_WPSignSuppressSupplemental><?xml version="1.0" encoding="utf-16"?>
<XStrings xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" /></CaseTypeDefinition_WPSignSuppressSupplemental>

<CaseTypeDefinition_NeverPreliminary>False</CaseTypeDefinition_NeverPreliminary>

<CaseTypeDefinition_AbnormalSpecialFlag>348386126</CaseTypeDefinition_AbnormalSpecialFlag>
```

```
<CaseTypeDefinition_WPSetSpecialFlag>True</CaseTypeDef
inition_WPSetSpecialFlag>

<CaseTypeDefinition_IgnoreLabelDefault>False</CaseType
Definition_IgnoreLabelDefault>
    <Constituent_TYPE>IMMUNO</Constituent_TYPE>

<Constituent_ObjectID>506873439</Constituent_ObjectID>
    <Constituent_CreationDate>2020-03-09T14:50:50.123-
04:00</Constituent_CreationDate>
    <Constituent_UpdateTime>2020-03-09T14:52:49.107-
04:00</Constituent_UpdateTime>

<Constituent_SuperobjectID>506873438</Constituent_Supe
robjectID>
    <Constituent_User1 />
    <Constituent_User2 />
    <Constituent_User3 />
    <Constituent_User4 />
    <Constituent_CREATOR>Scaron</Constituent_CREATOR>

<Constituent_MODIFIER>Scaron</Constituent_MODIFIER>

<Constituent_Reportable>True</Constituent_Reportable>
    <Constituent_Description />

<Constituent_Definition>414331635</Constituent_Definit
ion>
    <Constituent_Result>Negative</Constituent_Result>
    <Constituent_R1 />
    <Constituent_R2 />
    <Constituent_R3 />
    <Constituent_R4 />
    <Constituent_R5 />
    <Constituent_R6 />
    <Constituent_Abnormal>False</Constituent_Abnormal>
    <Constituent_AOV>False</Constituent_AOV>
    <Constituent_Range />
    <Constituent_RangeComment />

<Constituent_IsConfirm>False</Constituent_IsConfirm>

<Constituent_ConstituentDefinitionObjectID>414331635</
Constituent_ConstituentDefinitionObjectID>

<Constituent_ConstituentDefinitionCreationDate>2018-
10-26T11:55:05.557-
04:00</Constituent_ConstituentDefinitionCreationDate>
```

```
    <Constituent_ConstituentDefinitionUpdateTime>2020-11-03T10:33:26.493-05:00</Constituent_ConstituentDefinitionUpdateTime>

<Constituent_ConstituentDefinitionCREATOR>Import</Constituent_ConstituentDefinitionCREATOR>

<Constituent_ConstituentDefinitionMODIFIER>Scaron</Constituent_ConstituentDefinitionMODIFIER>

<Constituent_ConstituentDefinitionCode>AMPH</Constituent_ConstituentDefinitionCode>

<Constituent_ConstituentDefinitionDescription>Amphetamine</Constituent_ConstituentDefinitionDescription>

<Constituent_ConstituentDefinitionRETIRED>False</Constituent_ConstituentDefinitionRETIRED>
    <Constituent_ConstituentDefinitionCutOffValue>250 ng/dl</Constituent_ConstituentDefinitionCutOffValue>

<Constituent_ConstituentDefinitionDetectedString>DETECTED</Constituent_ConstituentDefinitionDetectedString>

<Constituent_ConstituentDefinitionUndetectedString>NOT DETECTED</Constituent_ConstituentDefinitionUndetectedString>

<Constituent_ConstituentDefinitionIllicit>False</Constituent_ConstituentDefinitionIllicit>

<Constituent_ConstituentDefinitionDetectionWindow>1-2 days</Constituent_ConstituentDefinitionDetectionWindow>

<Constituent_ConstituentDefinitionOrderConfirmationWhen>Detected</Constituent_ConstituentDefinitionOrderConfirmationWhen>

<Constituent_ConstituentDefinitionConfirmation>478867597</Constituent_ConstituentDefinitionConfirmation>

<Constituent_ConstituentDefinitionConfirmationCharge>259621897</Constituent_ConstituentDefinitionConfirmationCharge>

<Constituent_ConstituentDefinitionIsTextual>False</Constituent_ConstituentDefinitionIsTextual>
  </CaseTable>
</NewDataSet>
```

GET CASES

Endpoint	api/N/GetCases
Purpose	Returns a collection of NucleoLIS case records based on filter parameters passed.
Request Type	GET

Parameters	case_number={case_number}
	activesOnly={activesOnly}
	order_date_from={order_date_from}
	order_date_to={order_date_to}
	submitting_physicians={submitting_physicians}
	submitting_groups={submitting_groups}
	filter_expression={filter_expression}
	user_ID={user_ID}
	return_format={return_format}
	Where:
	case_number is the requested NucleoLIS case_number. (Wildcards are supported)
	activesOnly is a true or false value. If true, only active cases will be returned. If false, all cases will be returned.
	order_date_from is the order from date for filtering cases. Dates must be in the format YYYY-MM-DD.
	order_date_to is the order to date for filtering cases. Dates must be in the format YYYY-MM-DD.
	submitting_physicians is a list of submitting physicians.
	submitting_groups is a list of submitting groups.
	filter_expression is used as a secondary custom filtering criteria applied to the result set . Wildcards are supported. Any valid SQL WHERE clause is accepted. It can be used with any column returned in the result set. Multiple columns can be used when separated with 'AND'. Filtering criteria supported include: Patient_ObjectID, Patient_Name, Patient_Code, Patient_Dob,

	Patient_Sex, Case_Number, Case_OrderDate, Case_CreationDate, Case_ReportStatus, Case_Status, Case_ObjectID. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://SomeServer.com/api/N/GetCases?case_number=CL2020*&activesOnly=true&return_format=0&user_id=245654&order_date_from=2020-09-01&order_date_to=2020-11-02&filter_expression=case_status='A' AND patient_sex='Female'&submitting_physicians=*&submitting_groups=*
Returns	If the request succeeds, returns a collection of 1..n case records matching the filtering criteria in the following XML format: <?xml version="1.0" encoding="UTF-8" ?><NewDataSet> <CaseTable> <Patient_ObjectID>506408213</Patient_ObjectID> <Patient_Name>Boop, Betty</Patient_Name> <Patient_Code>506408213</Patient_Code> <Patient_Dob>1965-03-23</Patient_Dob> <Patient_Sex>Female</Patient_Sex> <Case_Number>CL2020-000080</Case_Number> <Case_OrderDate>2020-09-17</Case_OrderDate> <Case_CreationDate>2020-09-17T17:27:27.473-04:00</Case_CreationDate> <Case_ReportStatus>PENDING</Case_ReportStatus> <Case_Status>A</Case_Status> <Case_ObjectID>514638163</Case_ObjectID> </CaseTable> <CaseTable> <Patient_ObjectID>503799913</Patient_ObjectID> <Patient_Name>Bell, Tinker</Patient_Name> <Patient_Code>503799913</Patient_Code> <Patient_Dob>1950-01-01</Patient_Dob> <Patient_Sex>Female</Patient_Sex> <Case_Number>CL2020-000082</Case_Number> <Case_OrderDate>2020-09-18</Case_OrderDate> <Case_CreationDate>2020-09-18T14:53:08.89-04:00</Case_CreationDate> <Case_ReportStatus>PENDING</Case_ReportStatus> <Case_Status>A</Case_Status> <Case_ObjectID>514704625</Case_ObjectID> </CaseTable>

```
<CaseTable>
  <Patient_ObjectID>503799913</Patient_ObjectID>
  <Patient_Name>Bell, Tinker</Patient_Name>
  <Patient_Code>503799913</Patient_Code>
  <Patient_Dob>1950-01-01</Patient_Dob>
  <Patient_Sex>Female</Patient_Sex>
  <Case_Number>CL2020-000084</Case_Number>
  <Case_OrderDate>2020-09-28</Case_OrderDate>
  <Case_CreationDate>2020-09-28T13:43:54.22-04:00</Case_CreationDate>
  <Case_ReportStatus>PENDING</Case_ReportStatus>
  <Case_Status>A</Case_Status>
  <Case_ObjectID>515443274</Case_ObjectID>
</CaseTable>
<CaseTable>
  <Patient_ObjectID>503625913</Patient_ObjectID>
  <Patient_Name>Accession, Annie</Patient_Name>
  <Patient_Code>503625913</Patient_Code>
  <Patient_Dob>2018-12-11</Patient_Dob>
  <Patient_Sex>Female</Patient_Sex>
  <Case_Number>CL2020-000089</Case_Number>
  <Case_OrderDate>2020-09-30</Case_OrderDate>
  <Case_CreationDate>2020-09-30T13:32:44.8-04:00</Case_CreationDate>
  <Case_ReportStatus>PENDING</Case_ReportStatus>
  <Case_Status>A</Case_Status>
  <Case_ObjectID>515593040</Case_ObjectID>
</CaseTable>
<CaseTable>
  <Patient_ObjectID>433509601</Patient_ObjectID>
  <Patient_Name>Friday, Girl</Patient_Name>
  <Patient_Code>433509601</Patient_Code>
  <Patient_Dob>2009-03-13</Patient_Dob>
  <Patient_Sex>Female</Patient_Sex>
  <Case_Number>CL2020-000091</Case_Number>
  <Case_OrderDate>2020-10-05</Case_OrderDate>
  <Case_CreationDate>2020-10-05T15:56:57.673-04:00</Case_CreationDate>
  <Case_ReportStatus>PENDING</Case_ReportStatus>
  <Case_Status>A</Case_Status>
  <Case_ObjectID>515975948</Case_ObjectID>
</CaseTable>
<CaseTable>
  <Patient_ObjectID>516131995</Patient_ObjectID>
  <Patient_Name>Boop, Betty</Patient_Name>
  <Patient_Code>516131995</Patient_Code>
  <Patient_Dob>1965-03-23</Patient_Dob>
  <Patient_Sex>Female</Patient_Sex>
  <Case_Number>CL2020-000105</Case_Number>
  <Case_OrderDate>2020-10-07</Case_OrderDate>
  <Case_CreationDate>2020-10-07T14:49:44.863-04:00</Case_CreationDate>
```

```
    <Case_ReportStatus>PENDING</Case_ReportStatus>
    <Case_Status>A</Case_Status>
    <Case_ObjectID>516132019</Case_ObjectID>
  </CaseTable>
  <CaseTable>
    <Patient_ObjectID>516207223</Patient_ObjectID>
    <Patient_Name>Bell, Tinker</Patient_Name>
    <Patient_Code>516207223</Patient_Code>
    <Patient_Dob>1950-01-01</Patient_Dob>
    <Patient_Sex>Female</Patient_Sex>
    <Case_Number>CL2020-000107</Case_Number>
    <Case_OrderDate>2020-10-08</Case_OrderDate>
    <Case_CreationDate>2020-10-08T13:26:04.687-04:00</Case_CreationDate>
    <Case_ReportStatus>PRELIMINARY</Case_ReportStatus>
    <Case_Status>A</Case_Status>
    <Case_ObjectID>516207240</Case_ObjectID>
  </CaseTable>
  <CaseTable>
    <Patient_ObjectID>514701722</Patient_ObjectID>
    <Patient_Name>Boop, Betty</Patient_Name>
    <Patient_Code>514701722</Patient_Code>
    <Patient_Dob>1965-03-23</Patient_Dob>
    <Patient_Sex>Female</Patient_Sex>
    <Case_Number>CL2020-000109</Case_Number>
    <Case_OrderDate>2020-09-18</Case_OrderDate>
    <Case_CreationDate>2020-09-25T11:56:06.06-04:00</Case_CreationDate>
    <Case_ReportStatus>PRELIMINARY</Case_ReportStatus>
    <Case_Status>A</Case_Status>
    <Case_ObjectID>515221990</Case_ObjectID>
  </CaseTable>

</NewDataSet>
```

GET SPECIMEN

Endpoint	api/N/GetSpecimen
Purpose	Return a single, full NucleoLIS specimen record based on specimen unique identifier. Note: The full specimen record is inclusive of all specimen level data elements.
Request Type	GET

Parameters	`specimen_ID={specimen_ID}`
	`user_ID={user_ID}`

	`return_format={return_format}` Where: **specimen_id** is the specimen id number of the specimen to return. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	`https://www.SomeServer.com/api/N/GetSpecimen?specimen_id=467291&user_id=245654&return_format=0`

Returns	If the request succeeds, returns a specimen record matching the filtering criteria in the following XML format:
	`<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>`
	`  <SpecimenTable>`
	`    <Patient_ObjectID>347631</Patient_ObjectID>`
	`    <Patient_Name>Farrell, Ivy</Patient_Name>`
	`    <Patient_Code>347631</Patient_Code>`
	`    <Patient_Sex>F</Patient_Sex>`
	`    <Patient_Dob>1970-01-01</Patient_Dob>`
	`    <Case_Number>2</Case_Number>`
	`    <Case_Status>I</Case_Status>`
	`    <Specimen_ObjectID>467291</Specimen_ObjectID>`
	`    <Specimen_CreationDate>2015-03-27T15:56:44.83-04:00</Specimen_CreationDate>`
	`    <Specimen_UpdateTime>2015-03-30T07:07:02.71-04:00</Specimen_UpdateTime>`
	`<Specimen_SuperobjectID>467290</Specimen_SuperobjectID>`
	`    <Specimen_CREATOR>HIS</Specimen_CREATOR>`
	`    <Specimen_MODIFIER>0</Specimen_MODIFIER>`
	`    <Specimen_CollectionDate>2015-03-13</Specimen_CollectionDate>`
	`<Specimen_CollectionTime>16:14</Specimen_CollectionTime>`
	`    <Specimen_ReceivedDate>2015-03-30</Specimen_ReceivedDate>`
	`<Specimen_ReceivedTime>07:07:02</Specimen_ReceivedTime>`

	<Specimen_Number>467291</Specimen_Number> <Specimen_Status>A</Specimen_Status> <Specimen_Receiver>0</Specimen_Receiver> </SpecimenTable> </NewDataSet>

GET SPECIMENS

Endpoint	api/N/GetSpecimens
Purpose	Returns a collection of NucleoLIS Patient, Case and Specimen level data.
Request Type	GET

Parameters	received_date_from={received_date_from}
	received_date_to={received_date_to}
	activesOnly={activesOnly}
	specimen_status_steps={specimen_status_steps}
	specimen_sources={specimen_sources}
	filter_expression={filter_expression}
	user_ID={user_ID}
	return_format={return_format}
	Where:
	received_date_from is the "from" date for received specimens. Dates must by in YYYY-MM-DD format.
	received_date_to is the"to" date for received specimens. Dates must by in YYYY-MM-DD format.
	activesOnly is a true or false value. If true, only active cases are returned. If false, all cases are returned.
	specimen_status_steps is a pipe delimited list of specimen status steps.
	specimen_sources is a pipe delimited list of specimen sources
	filter_expression is used as a secondary custom filtering criteria applied to the result set . Wildcards are supported. It can be used with any column returned in the result set. Any valid SQL

	WHERE clause is accepted. Multiple columns can be used when separated with 'AND'. Filtering criteria supported include: Patient_ObjectID, Patient_Name, Patient_Code, Patient_Dob, Patient_Sex, Case_Number, Specimen_ObjectID, Specimen_Received date, Specimen_Received time, Specimen_CollectionDate, Specimen_CollectionTime, Specimen_StatusStep, SourceDefinition_Code, and SourceDefinition_Description. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/GetSpecimens?received_date_from=2020-10-01&received_date_to=2020-11-02&activesOnly=true&specimen_status_steps=&specimen_sources=Blood|Serum&filter_expression=Patient_Sex='Male'&user_id=245654&return_format=0
Returns	If the request succeeds, returns a collection of 1..n specimen records matching the filtering criteria in the following XML format: <?xml version="1.0" encoding="UTF-8" ?><NewDataSet> <SpecimenTable> <Patient_ObjectID>507179564</Patient_ObjectID> <Patient_Name>Bishop, Tom</Patient_Name> <Patient_Code>507179564</Patient_Code> <Patient_Dob>1963-08-02</Patient_Dob> <Patient_Sex>Male</Patient_Sex> <Case_Number>507179566OR</Case_Number> <Specimen_ReceivedDate>2020-10-13</Specimen_ReceivedDate> <Specimen_ReceivedTime>14:07:31</Specimen_ReceivedTime> <Specimen_CollectionDate>2020-10-13</Specimen_CollectionDate> <Specimen_CollectionTime /> <Specimen_StatusStep /> <SourceDefinition_Code>Blood</SourceDefinition_Code> <SourceDefinition_Description>Blood</SourceDefinition_Description>

```
    <Specimen_ObjectID>516628021</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>514700814</Patient_ObjectID>
    <Patient_Name>Barksalot, Bruno</Patient_Name>
    <Patient_Code>514700814</Patient_Code>
    <Patient_Dob>2016-03-30</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000086</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:15:58</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-09-
18</Specimen_CollectionDate>

<Specimen_CollectionTime>13:38</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>515221988</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>503782367</Patient_ObjectID>
    <Patient_Name>Bishop, Tom</Patient_Name>
    <Patient_Code>503782367</Patient_Code>
    <Patient_Dob>1963-08-02</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000093</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:15:34</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
06</Specimen_CollectionDate>
    <Specimen_CollectionTime />
    <Specimen_StatusStep />

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516056679</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>511955302</Patient_ObjectID>
    <Patient_Name>Cricket, Jimimy</Patient_Name>
    <Patient_Code>511955302</Patient_Code>
    <Patient_Dob>2018-11-13</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000110</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:16:15</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-07-
21</Specimen_CollectionDate>

<Specimen_CollectionTime>17:28</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>511968657</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>510788319</Patient_ObjectID>
    <Patient_Name>Caron, Jack</Patient_Name>
    <Patient_Code>510788319</Patient_Code>
    <Patient_Dob>2019-06-15</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000111</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:16:18</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-06-
08</Specimen_CollectionDate>

<Specimen_CollectionTime>16:20</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>510788325</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>506987816</Patient_ObjectID>
    <Patient_Name>Duck, Daffy</Patient_Name>
    <Patient_Code>506987816</Patient_Code>
    <Patient_Dob>1950-03-03</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000113</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:16:20</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-03-
12</Specimen_CollectionDate>

<Specimen_CollectionTime>14:40</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>506987822</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>506407748</Patient_ObjectID>
    <Patient_Name>Cricket, Jimimy</Patient_Name>
    <Patient_Code>506407748</Patient_Code>
    <Patient_Dob>2018-11-13</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000115</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:16:24</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-02-
28</Specimen_CollectionDate>

<Specimen_CollectionTime>14:57</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>506407755</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>298771370</Patient_ObjectID>
    <Patient_Name>Jones, Tom</Patient_Name>
    <Patient_Code>298771370</Patient_Code>
    <Patient_Dob>1944-01-01</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000155</Case_Number>
    <Specimen_ReceivedDate>2020-10-
21</Specimen_ReceivedDate>

<Specimen_ReceivedTime>17:10:05</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
21</Specimen_CollectionDate>
    <Specimen_CollectionTime />
    <Specimen_StatusStep />

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>517413942</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>517962619</Patient_ObjectID>
    <Patient_Name>Caron, Jack</Patient_Name>
    <Patient_Code>517962619</Patient_Code>
    <Patient_Dob>2019-06-15</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000156</Case_Number>
    <Specimen_ReceivedDate>2020-10-
27</Specimen_ReceivedDate>

<Specimen_ReceivedTime>10:50:07</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
27</Specimen_CollectionDate>

<Specimen_CollectionTime>10:46</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>517962628</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>518533883</Patient_ObjectID>
    <Patient_Name>Caron, Jack</Patient_Name>
    <Patient_Code>518533883</Patient_Code>
    <Patient_Dob>2019-06-15</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CL2020-000160</Case_Number>
    <Specimen_ReceivedDate>2020-11-
02</Specimen_ReceivedDate>
```

```
<Specimen_ReceivedTime>11:05:39</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-11-
02</Specimen_CollectionDate>

<Specimen_CollectionTime>10:59</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>518533891</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516213756</Patient_ObjectID>
    <Patient_Name>Caron, Jack</Patient_Name>
    <Patient_Code>516213756</Patient_Code>
    <Patient_Dob>2019-06-15</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000002</Case_Number>
    <Specimen_ReceivedDate>2020-10-
08</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:15:11</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
08</Specimen_CollectionDate>

<Specimen_CollectionTime>18:10</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516213763</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516287111</Patient_ObjectID>
    <Patient_Name>Caron, Jack</Patient_Name>
    <Patient_Code>516287111</Patient_Code>
    <Patient_Dob>2019-06-15</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000005</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>14:45:44</Specimen_ReceivedTim
e>
```

```
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>14:43</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516287120</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516287344</Patient_ObjectID>
    <Patient_Name>Cricket, Jimimy</Patient_Name>
    <Patient_Code>516287344</Patient_Code>
    <Patient_Dob>2018-11-13</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000006</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>15:44:18</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>14:46</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516287379</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516291416</Patient_ObjectID>
    <Patient_Name>Duck, Daffy</Patient_Name>
    <Patient_Code>516291416</Patient_Code>
    <Patient_Dob>1950-03-03</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000009</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:11:49</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>
```

```
<Specimen_CollectionTime>15:54</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516291421</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516276481</Patient_ObjectID>
    <Patient_Name>Duck, Daffy</Patient_Name>
    <Patient_Code>516276481</Patient_Code>
    <Patient_Dob>1950-03-03</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000010</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:12:16</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>11:46</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516276511</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516298160</Patient_ObjectID>
    <Patient_Name>QA, Final</Patient_Name>
    <Patient_Code>516298160</Patient_Code>
    <Patient_Dob>2019-10-23</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000013</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:13:24</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>17:45</Specimen_CollectionTi
me>
```

```
<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516298169</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>518078029</Patient_ObjectID>
    <Patient_Name>QA, Final</Patient_Name>
    <Patient_Code>518078029</Patient_Code>
    <Patient_Dob>2019-10-23</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>CLI2020-000016</Case_Number>
    <Specimen_ReceivedDate>2020-10-
28</Specimen_ReceivedDate>

<Specimen_ReceivedTime>15:48:59</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
28</Specimen_CollectionDate>

<Specimen_CollectionTime>15:44</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>518078227</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>510676219</Patient_ObjectID>
    <Patient_Name>Cricket, Jiminy</Patient_Name>
    <Patient_Code>510676219</Patient_Code>
    <Patient_Dob>2011-07-01</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>FC2020-000002</Case_Number>
    <Specimen_ReceivedDate>2020-10-
13</Specimen_ReceivedDate>

<Specimen_ReceivedTime>08:30:47</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-06-
04</Specimen_CollectionDate>

<Specimen_CollectionTime>15:26</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>
```

```
<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>510676225</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>516298310</Patient_ObjectID>
    <Patient_Name>QA, Final</Patient_Name>
    <Patient_Code>516298310</Patient_Code>
    <Patient_Dob>2019-10-23</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>HE2020-000003</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:14:16</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-10-
09</Specimen_CollectionDate>

<Specimen_CollectionTime>17:49</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
    <Specimen_ObjectID>516298330</Specimen_ObjectID>
  </SpecimenTable>
  <SpecimenTable>
    <Patient_ObjectID>506647493</Patient_ObjectID>
    <Patient_Name>Mouse, Mickey</Patient_Name>
    <Patient_Code>506647493</Patient_Code>
    <Patient_Dob>2000-07-08</Patient_Dob>
    <Patient_Sex>Male</Patient_Sex>
    <Case_Number>M2020-000030</Case_Number>
    <Specimen_ReceivedDate>2020-10-
09</Specimen_ReceivedDate>

<Specimen_ReceivedTime>18:16:21</Specimen_ReceivedTim
e>
    <Specimen_CollectionDate>2020-03-
04</Specimen_CollectionDate>

<Specimen_CollectionTime>16:09</Specimen_CollectionTi
me>

<SourceDefinition_Code>Blood</SourceDefinition_Code>

<SourceDefinition_Description>Blood</SourceDefinition
_Description>
```

```
     <Specimen_ObjectID>506647513</Specimen_ObjectID>
   </SpecimenTable>
   <SpecimenTable>
     <Patient_ObjectID>510588219</Patient_ObjectID>
     <Patient_Name>Covid, Charlie</Patient_Name>
     <Patient_Code>510588219</Patient_Code>
     <Patient_Dob>2019-12-15</Patient_Dob>
     <Patient_Sex>Male</Patient_Sex>
     <Case_Number>M2020-000059</Case_Number>
     <Specimen_ReceivedDate>2020-10-
 09</Specimen_ReceivedDate>

 <Specimen_ReceivedTime>18:36:05</Specimen_ReceivedTim
 e>
     <Specimen_CollectionDate>2020-06-
 01</Specimen_CollectionDate>

 <Specimen_CollectionTime>10:00</Specimen_CollectionTi
 me>
     <Specimen_StatusStep />

 <SourceDefinition_Code>Blood</SourceDefinition_Code>

 <SourceDefinition_Description>Blood</SourceDefinition
 _Description>
     <Specimen_ObjectID>510589737</Specimen_ObjectID>
   </SpecimenTable>
</NewDataSet>
```

GET TEST ORDER

Endpoint	api/N/GetTestOrder
Purpose	Returns a single, full test record and available results based on test order unique identifier.
Request Type	GET

Parameters	`test_order_ID={test_order_ID}`
	`user_ID={user_ID}`
	`return_format={return_format}`
	Where:

	test_order_id is the id of the test order to return. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/GetTestOrder?test_order_id=8606534&user_id=245654&return_format=0
Returns	If the request succeeds, it returns the requested full profile test order record inclusive of all test order and available test result data in the following XML format: <?xml version="1.0" encoding="UTF-8" ?><NewDataSet> <ProfileTable> <Patient_ObjectID>4180414</Patient_ObjectID> <Patient_Name>Wednesday, Wendy</Patient_Name> <Patient_Code>4180414</Patient_Code> <Patient_Sex /> <Patient_Dob /> <Case_Number>PSY2015-000042</Case_Number> <Case_Status>I</Case_Status> <Specimen_ObjectID>8606398</Specimen_ObjectID> <Specimen_CollectionDate /> <Specimen_CollectionTime /> <Specimen_ReceivedDate>2015-05-19</Specimen_ReceivedDate> <Specimen_ReceivedTime>15:37:45</Specimen_ReceivedTime> <Specimen_StatusStep /> <SourceDefinition_Code>Blood</SourceDefinition_Code> <SourceDefinition_Description>Blood</SourceDefinition_Description> <Profile_TYPE>FISH</Profile_TYPE> <Profile_ObjectID>8606534</Profile_ObjectID> <Profile_CreationDate>2015-05-19T15:37:18.65-04:00</Profile_CreationDate> <Profile_UpdateTime>2015-05-19T15:42:05.01-04:00</Profile_UpdateTime>

```
<Profile_SuperobjectID>8606398</Profile_SuperobjectID
>
    <Profile_User1 />
    <Profile_User2 />
    <Profile_User3 />
    <Profile_User4 />
    <Profile_CREATOR>deannac</Profile_CREATOR>
    <Profile_MODIFIER>deannac</Profile_MODIFIER>
    <Profile_Description />
    <Profile_OrderDate>2015-05-19</Profile_OrderDate>
    <Profile_OrderTime>15:37</Profile_OrderTime>
    <Profile_OrderedBy>deannac</Profile_OrderedBy>

<Profile_PanelDefinition>249815</Profile_PanelDefinit
ion>
    <Profile_R1>Clinical Significance: ROS1 gene
rearrangements are found in 1-2% of non-small cell
lung carcinoma (NSCLC). Pre-clinical and early
clinical evidence suggests ROS1-rearranged tumors may
be sensitive to the dual ALK/MET inhibitor
crizotinib.</Profile_R1>
    <Profile_R2 />
    <Profile_R3 />
    <Profile_R4 />
    <Profile_R5 />
    <Profile_R6 />
    <Profile_Result>Present</Profile_Result>
    <Profile_Reportable>True</Profile_Reportable>
    <Profile_DatumDef>245550</Profile_DatumDef>

<Profile_DatumEntryDefinition>4380952</Profile_DatumE
ntryDefinition>
    <Profile_Images><?xml version="1.0"
encoding="utf-16"?>
<ChrysalisImgDats
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ImgDat />
</ChrysalisImgDats></Profile_Images>
    <Profile_Abnormal>False</Profile_Abnormal>
    <Profile_AOV>False</Profile_AOV>
    <Profile_StatusStep />
    <Profile_Range />
    <Profile_RangeComment />
    <Profile_CanceledDate />
    <Profile_CanceledBy />
    <Profile_CanceledFor />
    <Profile_ReflexedBy />
    <Profile_IOrder />
    <Profile_IOrderDate />
    <Profile_IOrderTime />
    <Profile_Complete>True</Profile_Complete>

<Profile_RequireSignature>false</Profile_RequireSigna
ture>

<Profile_RequireApproval>false</Profile_RequireApprov
al>

<Profile_ProfileDefinitionObjectID>249815</Profile_Pr
ofileDefinitionObjectID>
    <Profile_ProfileDefinitionCreationDate>2014-12-
31T10:55:34.78-
05:00</Profile_ProfileDefinitionCreationDate>
    <Profile_ProfileDefinitionUpdateTime>2015-10-
01T15:08:20.637-
04:00</Profile_ProfileDefinitionUpdateTime>
    <Profile_ProfileDefinitionUser1 />
    <Profile_ProfileDefinitionUser2 />
    <Profile_ProfileDefinitionUser3 />
    <Profile_ProfileDefinitionUser4 />

<Profile_ProfileDefinitionCREATOR>0</Profile_ProfileD
efinitionCREATOR>

<Profile_ProfileDefinitionMODIFIER>deannac</Profile_P
rofileDefinitionMODIFIER>

<Profile_ProfileDefinitionCode>NSmCCA</Profile_Profil
eDefinitionCode>
    <Profile_ProfileDefinitionDescription>Non-Small
Cell Carcinoma,
Lung</Profile_ProfileDefinitionDescription>

<Profile_ProfileDefinitionRETIRED>False</Profile_Prof
ileDefinitionRETIRED>

<Profile_ProfileDefinitionDatumLink>245550</Profile_P
rofileDefinitionDatumLink>

<Profile_ProfileDefinitionConstituentDatumLink>342477
12</Profile_ProfileDefinitionConstituentDatumLink>

<Profile_ProfileDefinitionDatumEntryDefinition>347149
41</Profile_ProfileDefinitionDatumEntryDefinition>

<Profile_ProfileDefinitionAlwaysReportMessage>ROS1
FISH uses a dual-color break-apart probe which
produces single green, red, and yellow or fusion
signals in rearranged cells.
```

```
References:
    Kwak EL et al. New Engl J Med. 2010;363(18):1693-
1703.
    Vysis ALK Break Apart FISH Probe Kit package
insert. Vysis ALK United States website. Accessed 12-
31-2012.
    Bergethon K, Shaw AT, Ignatius Ou S, et al. J
Clin Oncol. 2012;30(8):863-870.
    Takeuchi K, Soda M, Togashi Y, et al. Nature Med.
2012;18(3):378-381.
    Davies KD, Le AT, Theodoro MF, et al. Clin Cancer
Res. 2012;18(17):4570-4579.
    Shaw AT, Camidge DR, Engelman JA, et al. J Clin
Oncol. 2012;30 (suppl): abstract 7508.
    Bang YJ. Arch Pathol Lab Med. 2012;136:1201-1204.
    NCCN Guidelines, Non-Small Cell Lung Cancer,
version 1.2013. National Comprehensive Cancer Network
website. Accessed 12-31-2012.
    ALK page. Atlas of Genetics and Cytogenetics in
Oncology and Haematology. Accessed 12-31-2012; last
updated 2-2010.
    Stumpfova M, Janne PA. Clin Cancer Res.
2012;18(16):4222-4224.
    Sasaki T et al. Eur J Cancer. 2010;46:1773-1780.
</Profile_ProfileDefinitionAlwaysReportMessage>

<Profile_ProfileDefinitionMethodology>FISH</Profile_P
rofileDefinitionMethodology>
    <Profile_ProfileDefinitionCodeMap />

<Profile_ProfileDefinitionLabelCount>1</Profile_Profi
leDefinitionLabelCount>

<Profile_ProfileDefinitionAutoComplete>False</Profile
_ProfileDefinitionAutoComplete>

<Profile_ProfileDefinitionWPDiscreteAdvanceOnly>false
</Profile_ProfileDefinitionWPDiscreteAdvanceOnly>

<Profile_DatumDefinitionObjectID>245550</Profile_Datu
mDefinitionObjectID>
    <Profile_DatumDefinitionCreationDate>2014-12-
31T10:54:49.03-
05:00</Profile_DatumDefinitionCreationDate>
    <Profile_DatumDefinitionUpdateTime>2019-04-
09T15:17:02.137-
04:00</Profile_DatumDefinitionUpdateTime>
    <Profile_DatumDefinitionUser1 />
    <Profile_DatumDefinitionUser2 />
    <Profile_DatumDefinitionUser3 />
    <Profile_DatumDefinitionUser4 />
```

```
<Profile_DatumDefinitionCREATOR>0</Profile_DatumDefin
itionCREATOR>

<Profile_DatumDefinitionMODIFIER>0</Profile_DatumDefi
nitionMODIFIER>

<Profile_DatumDefinitionCode>416980</Profile_DatumDef
initionCode>
    <Profile_DatumDefinitionDescription />

<Profile_DatumDefinitionRETIRED>False</Profile_DatumD
efinitionRETIRED>

<Profile_DatumDefinitionR1>Comment</Profile_DatumDefi
nitionR1>
    <Profile_DatumDefinitionR2 />
    <Profile_DatumDefinitionR3 />
    <Profile_DatumDefinitionR4 />
    <Profile_DatumDefinitionR5 />
    <Profile_DatumDefinitionR6 />

<Profile_DatumDefinitionHideResult>False</Profile_Dat
umDefinitionHideResult>
    <Profile_DatumDefinitionRange />
    <Profile_DatumDefinitionRangeComment />
    <Profile_DatumDefinitionExtensions><?xml
version="1.0" encoding="utf-16"?>
<DatumExtensionDefinition
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Items>
    <DatumExtensionItemDefinition>
      <FieldName>Result</FieldName>
      <IsReadOnly>false</IsReadOnly>
    </DatumExtensionItemDefinition>
    <DatumExtensionItemDefinition>
      <FieldName>R1</FieldName>
      <IsReadOnly>false</IsReadOnly>
    </DatumExtensionItemDefinition>
    <DatumExtensionItemDefinition>
      <FieldName>R2</FieldName>
      <IsReadOnly>false</IsReadOnly>
    </DatumExtensionItemDefinition>
    <DatumExtensionItemDefinition>
      <FieldName>R3</FieldName>
      <IsReadOnly>false</IsReadOnly>
    </DatumExtensionItemDefinition>
    <DatumExtensionItemDefinition>
      <FieldName>R4</FieldName>
      <IsReadOnly>false</IsReadOnly>
      </DatumExtensionItemDefinition>
      <DatumExtensionItemDefinition>
        <FieldName>R5</FieldName>
        <IsReadOnly>false</IsReadOnly>
      </DatumExtensionItemDefinition>
      <DatumExtensionItemDefinition>
        <FieldName>R6</FieldName>
        <IsReadOnly>false</IsReadOnly>
      </DatumExtensionItemDefinition>
      <DatumExtensionItemDefinition>
        <FieldName>Range</FieldName>
        <IsReadOnly>true</IsReadOnly>
      </DatumExtensionItemDefinition>
      <DatumExtensionItemDefinition>
        <FieldName>RangeComment</FieldName>
        <IsReadOnly>true</IsReadOnly>
      </DatumExtensionItemDefinition>
    </Items>
  </DatumExtensionDefinition></Profile_DatumDefinitionE
  xtensions>
      <Profile_DatumDefinitionTableDisplayDefinition />
      <Profile_DatumDefinitionResultFieldMap />
      <Profile_DatumDefinitionInstrumentAutoOrder />
      <Constituent_TYPE>FISH</Constituent_TYPE>

  <Constituent_ObjectID>8606535</Constituent_ObjectID>
      <Constituent_CreationDate>2015-05-
  19T15:37:18.843-04:00</Constituent_CreationDate>
      <Constituent_UpdateTime>2015-05-19T15:39:04.96-
  04:00</Constituent_UpdateTime>

  <Constituent_SuperobjectID>8606534</Constituent_Super
  objectID>
      <Constituent_User1 />
      <Constituent_User2 />
      <Constituent_User3 />
      <Constituent_User4 />

  <Constituent_CREATOR>deannac</Constituent_CREATOR>

  <Constituent_MODIFIER>deannac</Constituent_MODIFIER>

  <Constituent_Reportable>True</Constituent_Reportable>
      <Constituent_Description />

  <Constituent_Definition>249789</Constituent_Definitio
  n>
      <Constituent_Result>Present</Constituent_Result>
      <Constituent_R1 />
      <Constituent_R2 />
      <Constituent_R3 />
    <Constituent_R4 />
    <Constituent_R5 />
    <Constituent_R6 />

<Constituent_Abnormal>False</Constituent_Abnormal>
    <Constituent_AOV>False</Constituent_AOV>
    <Constituent_Range />
    <Constituent_RangeComment />

<Constituent_ConstituentDefinitionObjectID>249789</Co
nstituent_ConstituentDefinitionObjectID>

<Constituent_ConstituentDefinitionCreationDate>2014-
12-31T10:55:34.527-
05:00</Constituent_ConstituentDefinitionCreationDate>

<Constituent_ConstituentDefinitionUpdateTime>2014-12-
31T10:55:34.527-
05:00</Constituent_ConstituentDefinitionUpdateTime>
    <Constituent_ConstituentDefinitionUser1 />
    <Constituent_ConstituentDefinitionUser2 />
    <Constituent_ConstituentDefinitionUser3 />
    <Constituent_ConstituentDefinitionUser4 />

<Constituent_ConstituentDefinitionCREATOR>0</Constitu
ent_ConstituentDefinitionCREATOR>

<Constituent_ConstituentDefinitionMODIFIER>0</Constit
uent_ConstituentDefinitionMODIFIER>

<Constituent_ConstituentDefinitionCode>ROS1</Constitu
ent_ConstituentDefinitionCode>

<Constituent_ConstituentDefinitionDescription>break-
apart probe at
(6q22)</Constituent_ConstituentDefinitionDescription>

<Constituent_ConstituentDefinitionRETIRED>False</Cons
tituent_ConstituentDefinitionRETIRED>
    <Constituent_ConstituentDefinitionCodeMap />
  </ProfileTable>
</NewDataSet>
```

GET TEST ORDERS

Endpoint	api/N/GetTestOrders

Purpose	Returns a collection of Patient, Case, Specimen and Test level data including Patient_ObjectID, Patient_Name, Patient_Code, Patient_Dob, Patient_Sex, Case_Number, Specimen_ObjectID, Specimen_Received date, Specimen_Received time, Specimen_CollectionDate, Specimen_CollectionTime, Specimen_StatusStep, SourceDefinition_Code, SourceDefinition_Description, Test order date, status, completion status, test code and unique identifier of all ordered tests records meeting filtering requirements. .
Request Type	GET

Parameters	`order_date_from={order_date_from}`
	`order_date_to={order_date_to}`
	`activesOnly={activesOnly}`
	`order_status_steps={order_status_steps}`
	`order_codes={order_codes}`
	`filter_expression={filter_expression}`
	`user_ID={user_ID}`
	`return_format={return_format}`
	Where:
	order_date_from is the “from” order date. Dates must be in YYYY-MM-DD format.
	order_date_to is the “to” order date. Dates must be in YYYY-MM-DD format.
	activesOnly is a true or false value. If set to true, only active orders records are returned if false, all order records are returned.
	order_status_steps is pipe delimited list of order status steps
	order_codes is a pipe delimited list of order codes
	filter_expression is used as a secondary custom filtering criteria applied to the result set . Wildcards are supported. It can be used with any column returned in the result set. Any valid SQL WHERE clause is accepted. Multiple columns can be used when separated with ‘AND’. Filtering criteria supported include: Patient_ObjectID, Patient_Name, Patient_Code, Patient_DOB, Patient_Sex, Case_Number, Specimen_ObjectID, Specimen_RetreivalDate, Specimen_RetreivalTime,

	Specimen_CollectionDate, Specimen_CollectionTime, Specimen_StatusStep, SourceDefinition_Code, SourceDefinition_Description, Profile_ObjectID, Profile_OrderDate, Profile_OrderTime, Profile_ProfileDefinitionCode, Profile_ProfileDefinitionDescription, Profile_Complete, Profile_CompleteDate, Profile_SignerDate, and Profile_StatusStep. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	`https://www.SomeServer.com/api/N/GetTestOrders?order_date_from=2020-09-17&order_date_to=2020-09-18&activesOnly=true&order_status_steps=*&order_codes=CDCI&filter_expression=Patient_DOB like '1960%'&user_ID=245654&return_format=0`
Returns	If the request succeeds, returns a collection of 1..n test order profile records matching the filtering criteria in the following XML format: <?xml version="1.0" encoding="UTF-8" ?><NewDataSet> <ProfileTable> <Patient_ObjectID>506646654</Patient_ObjectID> <Patient_Name>Buckley, Betty Sue Joe</Patient_Name> <Patient_Code>506646654</Patient_Code> <Patient_Dob>1960-07-14</Patient_Dob> <Patient_Sex>Female</Patient_Sex> <Case_Number>CL2020-000013</Case_Number> <Specimen_ObjectID>506686900</Specimen_ObjectID> <Specimen_ReceivedDate>2020-03-04</Specimen_ReceivedDate> <Specimen_ReceivedTime>15:49:27</Specimen_ReceivedTime> <Specimen_CollectionDate>2020-03-04</Specimen_CollectionDate> <Specimen_CollectionTime>15:46</Specimen_CollectionTime> <Specimen_StatusStep /> <SourceDefinition_Code>Blood</SourceDefinition_Code> <SourceDefinition_Description>Blood</SourceDefinition_Description> <Profile_ObjectID>514633950</Profile_ObjectID> <Profile_OrderDate>2020-09-17</Profile_OrderDate> <Profile_OrderTime>16:05</Profile_OrderTime> <Profile_ProfileDefinitionCode>CDCI</Profile_ProfileDefinitionCode>

```
<Profile_ProfileDefinitionDescription>CDCI</Profile_ProfileDefinitionDescription>
  <Profile_Complete>False</Profile_Complete>
  <Profile_CompleteDate />
  <Profile_SignerDate />
  <Profile_StatusStep />
 </ProfileTable>
 <ProfileTable>
  <Patient_ObjectID>291217280</Patient_ObjectID>
  <Patient_Name>Ashbey, Clywd</Patient_Name>
  <Patient_Code>291217280</Patient_Code>
  <Patient_Dob>1960-07-07</Patient_Dob>
  <Patient_Sex>Male</Patient_Sex>
  <Case_Number>CL2020-000068</Case_Number>
  <Specimen_ObjectID>511969438</Specimen_ObjectID>
  <Specimen_ReceivedDate>2020-08-06</Specimen_ReceivedDate>
  <Specimen_ReceivedTime>08:22:09</Specimen_ReceivedTime>
  <Specimen_CollectionDate>2020-08-06</Specimen_CollectionDate>
  <Specimen_CollectionTime />
  <Specimen_StatusStep />
  <SourceDefinition_Code>Blood</SourceDefinition_Code>
  <SourceDefinition_Description>Blood</SourceDefinition_Description>
  <Profile_ObjectID>514715461</Profile_ObjectID>
  <Profile_OrderDate>2020-09-18</Profile_OrderDate>
  <Profile_OrderTime>18:21</Profile_OrderTime>
  <Profile_ProfileDefinitionCode>CDCI</Profile_ProfileDefinitionCode>

<Profile_ProfileDefinitionDescription>CDCI</Profile_ProfileDefinitionDescription>
  <Profile_Complete>False</Profile_Complete>
  <Profile_CompleteDate />
  <Profile_SignerDate />
  <Profile_StatusStep />
 </ProfileTable>
 <ProfileTable>
  <Patient_ObjectID>132099125</Patient_ObjectID>
  <Patient_Name>Miller, Leonard</Patient_Name>
  <Patient_Code>132099125</Patient_Code>
  <Patient_Dob>1960-12-12</Patient_Dob>
  <Patient_Sex>Male</Patient_Sex>
  <Case_Number>CL2020-000079</Case_Number>
  <Specimen_ObjectID>514704804</Specimen_ObjectID>
  <Specimen_ReceivedDate>2020-09-18</Specimen_ReceivedDate>
  <Specimen_ReceivedTime>16:57:21</Specimen_ReceivedTime>
  <Specimen_CollectionDate>2020-09-18</Specimen_CollectionDate>
  <Specimen_CollectionTime />
  <Specimen_StatusStep />
```

```
    <SourceDefinition_Code>Blood</SourceDefinition_Code>
    <SourceDefinition_Description>Blood</SourceDefinition_Description>
    <Profile_ObjectID>514704809</Profile_ObjectID>
    <Profile_OrderDate>2020-09-18</Profile_OrderDate>
    <Profile_OrderTime>14:56</Profile_OrderTime>
    <Profile_ProfileDefinitionCode>CDCI</Profile_ProfileDefinitionCode>

<Profile_ProfileDefinitionDescription>CDCI</Profile_ProfileDefinitionDescript
ion>
    <Profile_Complete>False</Profile_Complete>
    <Profile_CompleteDate />
    <Profile_SignerDate />
    <Profile_StatusStep />
  </ProfileTable>
  <ProfileTable>
    <Patient_ObjectID>376500</Patient_ObjectID>
    <Patient_Name>Farrell, Tiffany</Patient_Name>
    <Patient_Code>376500</Patient_Code>
    <Patient_Dob>1960-07-01</Patient_Dob>
    <Patient_Sex>Female</Patient_Sex>
    <Case_Number>PSY2015-000029</Case_Number>
    <Specimen_ObjectID>4350748</Specimen_ObjectID>
    <Specimen_ReceivedDate>2015-06-02</Specimen_ReceivedDate>
    <Specimen_ReceivedTime>07:40:06</Specimen_ReceivedTime>
    <Specimen_CollectionDate>2015-04-22</Specimen_CollectionDate>
    <Specimen_CollectionTime>10:00</Specimen_CollectionTime>
    <Specimen_StatusStep />
    <SourceDefinition_Code>Blood</SourceDefinition_Code>
    <SourceDefinition_Description>Blood</SourceDefinition_Description>
    <Profile_ObjectID>514703397</Profile_ObjectID>
    <Profile_OrderDate>2020-09-18</Profile_OrderDate>
    <Profile_OrderTime>14:30</Profile_OrderTime>
    <Profile_ProfileDefinitionCode>CDCI</Profile_ProfileDefinitionCode>

<Profile_ProfileDefinitionDescription>CDCI</Profile_ProfileDefinitionDescript
ion>
    <Profile_Complete>False</Profile_Complete>
    <Profile_CompleteDate />
    <Profile_SignerDate />
    <Profile_StatusStep />
  </ProfileTable>
</NewDataSet>
```

GET PHYSICIAN

Endpoint	api/N/GetPhysician
Purpose	Returns a single, full physician record based on a unique code.
Request Type	GET
Parameters	physician_code={physician_code} user_ID={user_ID} return_format={return_format} Where: **physician_code** is the unique identifier for the requested physician. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/ api/N/GetPhysician?physician_code=381596&user_ID=245654&return_format=0

Returns	If the request succeeds, returns the requested physician record in the following format:
	<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
	  <PhysicianTable>
	    <Physician_ObjectID>230022</Physician_ObjectID>
	    <Physician_CreationDate>2004-07-30T12:14:57.687-04:00</Physician_CreationDate>
	    <Physician_UpdateTime>2020-01-21T11:25:23.293-05:00</Physician_UpdateTime>
	    <Physician_Name>test, physician</Physician_Name>
	    <Physician_Sex />
	    <Physician_Street />
	    <Physician_City />
	    <Physician_State />
	    <Physician_Zip />
	    <Physician_Phone />
	    <Physician_Code>TEST1</Physician_Code>

```
    <Physician_Soundex>TESTPHYSCN</Physician_Soundex>
    <Physician_UPIN />
    <Physician_STOP />
    <Physician_STREET2 />
    <Physician_RETIRED>False</Physician_RETIRED>
    <Physician_GROUP>False</Physician_GROUP>
    <Physician_STREET3 />
    <Physician_MODIFIER>0</Physician_MODIFIER>
    <Physician_NPI />
    <Physician_WebPathStartDate />

<Physician_WithholdGroupResults>False</Physician_With
holdGroupResults>
    <Physician_WebPathReturnAs />
    <Physician_WebCopyTo>False</Physician_WebCopyTo>
    <Physician_Instructions />

<Physician_WebPathSupP>False</Physician_WebPathSupP>

<Physician_WebPathALL>False</Physician_WebPathALL>
    <Physician_WebPathReturnCC><ALL MY
GROUPS></Physician_WebPathReturnCC>

<Physician_UsePreliminaryDistribution>False</Physicia
n_UsePreliminaryDistribution>
    <Physician_PreliminaryReportDistribution><?xml
version="1.0" encoding="utf-16"?>
<DistributionDefinition
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <AsSubmtterDistribution>
    <PrintEnabled>false</PrintEnabled>
    <FaxEnabled>false</FaxEnabled>
    <SecureMailEnabled>false</SecureMailEnabled>
  </AsSubmtterDistribution>
  <AsOtherDistribution>
    <PrintEnabled>false</PrintEnabled>
    <FaxEnabled>false</FaxEnabled>
    <SecureMailEnabled>false</SecureMailEnabled>
  </AsOtherDistribution>
</DistributionDefinition></Physician_PreliminaryRepor
tDistribution>
  </PhysicianTable>
</NewDataSet>
```

GET PHYSICIANS

Endpoint	api/N/GetPhysicians
Purpose	Returns a collection of 1..n physician records based on filter parameters passed.
Request Type	GET
Parameters	last_name={last_name} first_name={first_name} filter_expression={filter_expression} user_id={user_id} return_format={return_format} Where: **last_name** is the last name of the Physician. Wildcards are supported. **first_name** is the first name of the physician. Wildcards are supported. **filter_expression** is is used as a secondary custom filtering criteria applied to the result set . Wildcards are supported. Any valid SQL WHERE clause is accepted. It can be used with any column returned in the result set. Multiple columns can be used when separated with 'AND'. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the record. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/GetPhysicians?last_name=*&first_name=T*&filter_expression=Physician_GROUP='false'&user_id=245654&return_format=0
Returns	If the request succeeds, returns the requested physician records in the following format: <?xml version="1.0" encoding="UTF-8" ?><NewDataSet> <PhysicianTable>

```
    <Physician_ObjectID>15460945</Physician_ObjectID>
    <Physician_Name>Jones, Tim</Physician_Name>
    <Physician_Street>This physician has a single
"none" location called Office</Physician_Street>
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>JONES</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>

<Physician_ObjectID>503790496</Physician_ObjectID>
    <Physician_Name>New physician, testing
roles</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>NP1</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>

<Physician_ObjectID>503612491</Physician_ObjectID>
    <Physician_Name>Sig, Text Only</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>Phy01</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>
    <Physician_ObjectID>390657</Physician_ObjectID>
    <Physician_Name>Physician, Test</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>PHY2</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>
    <Physician_ObjectID>14661001</Physician_ObjectID>
    <Physician_Name>Physician, Test</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>PHY3</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>
    <Physician_ObjectID>16932034</Physician_ObjectID>
    <Physician_Name>Physician,
TestAgain</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>PHY5</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>

<Physician_ObjectID>135677272</Physician_ObjectID>
    <Physician_Name>Robins, Tim</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>ROBINS</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
  <PhysicianTable>

<Physician_ObjectID>232073067</Physician_ObjectID>
    <Physician_Name>Robins, Tim Psy</Physician_Name>
    <Physician_Street />
    <Physician_STREET2 />
    <Physician_STREET3 />
    <Physician_City />
    <Physician_State />
    <Physician_Zip />
    <Physician_Code>ROBINS/PSY</Physician_Code>
    <Physician_GROUP>False</Physician_GROUP>
  </PhysicianTable>
```

```
   <PhysicianTable>

 <Physician_ObjectID>135676995</Physician_ObjectID>
     <Physician_Name>Smith, Tim</Physician_Name>
     <Physician_Street />
     <Physician_STREET2 />
     <Physician_STREET3 />
     <Physician_City />
     <Physician_State />
     <Physician_Zip />
     <Physician_Code>SMITH</Physician_Code>
     <Physician_GROUP>False</Physician_GROUP>
   </PhysicianTable>
   <PhysicianTable>

 <Physician_ObjectID>232073138</Physician_ObjectID>
     <Physician_Name>Smith, Tim Psy</Physician_Name>
     <Physician_Street />
     <Physician_STREET2 />
     <Physician_STREET3 />
     <Physician_City />
     <Physician_State />
     <Physician_Zip />
     <Physician_Code>SMITH/PSY</Physician_Code>
     <Physician_GROUP>False</Physician_GROUP>
   </PhysicianTable>
   <PhysicianTable>

 <Physician_ObjectID>119430985</Physician_ObjectID>
     <Physician_Name>Doctor, Thursday</Physician_Name>
     <Physician_Street />
     <Physician_STREET2 />
     <Physician_STREET3 />
     <Physician_City />
     <Physician_State />
     <Physician_Zip />
     <Physician_Code>THURS</Physician_Code>
     <Physician_GROUP>False</Physician_GROUP>
   </PhysicianTable>
</NewDataSet>
```

SET STATUS STEP

Endpoint	api/N/ SetStatusStep
Purpose	Set a processing status of a single case, specimen or test order record.

Request Type	GET
Parameters	object_id={object_id} status_set={status_set} status_advance={status_advance} user_ID={user_ID} return_format={return_format} Where: **object_id** is the objectId of the record to update. **status_set** is the new status setting. **status_advance** is a true or false value. If true, the status of the record to update will be advanced to the next defined step. When false, the status_advance argument is ignored. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the status change. **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/SetStatusStep?object_id=503766463&status_set=&status_advance=true&user_ID=200001&return_format=0

Returns	If the request succeeds, returns the results of processing the status update:
	<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
	  <StatusStepUpdateTable>
	    <object_id>503766463</object_id>
	    <object_type>PCR</object_type>
	    <status_set>Purification</status_set>
	    <original_status>Accessioned</original_status>
	    <message>success</message>
	  </StatusStepUpdateTable>
	</NewDataSet>

SET STATUS STEPS

Endpoint	api/N/SetStatusSteps
Purpose	Set a processing status of a multiple case, specimen or test order records.
Request Type	GET
Parameters	object_ids={object_ids} status_set={status_set} status_advance={status_advance} user_ID={user_ID} return_format={return_format} Where: **object_ids** is a pipe delimited list of object ids that specifies which cases should be updated. **status_set** is the new processing status used to update the specified cases. **status_advance** is a true or false value. If true, the specified cases will be advanced. If false the specified cases will not be advanced. **user_id** is the user code for the user in the NucleoLIS user table that is requesting the status change(s). **return_format** is the format for returning the data. Where: 0 = XML Format 1 = JSON Format
Example	https://www.SomeServer.com/api/N/SetStatusSteps?object_ids=504252001|504165951&status_set=Amplify&status_advance=false&user_ID=200001&return_format=0

Returns	If the request succeeds, returns the results of processing the status updates:
	
	<?xml version="1.0" encoding="UTF-8" ?><NewDataSet>
	  <StatusStepUpdateTable>
	    <object_id>504252001</object_id>
	    <object_type>PCR</object_type>

```
    <status_set>Amplify</status_set>
    <original_status>Accessioned</original_status>
    <message>success</message>
  </StatusStepUpdateTable>
  <StatusStepUpdateTable>
    <object_id>504165951</object_id>
    <object_type>PCR</object_type>
    <status_set>Amplify</status_set>
    <original_status>Accessioned</original_status>
    <message>success</message>
  </StatusStepUpdateTable>
</NewDataSet>
```

APPENDIX B – CODING EXAMPLES

CODING EXAMPLES INTRODUCTION

Note: The coding examples in this appendix assume that the URL being connected to is https://www.SomeServer.com, and the username = LAB, and the password = 123456. These examples show how to connect to the Backbone API using a variety of programming languages and techniques. These examples are solely to help you get started communicating with the API, and are not intended to be used as production code.

For more information on the various frameworks mentioned in the code examples, please refer to the links below:

Language	**Framework**
C	https://curl.haxx.se/libcurl/c/
C#	http://restsharp.org/
cURL	https://curl.haxx.se/
Go	https://golang.org/pkg/net/http/
HTTP	None (Raw HTTP request)
Java OkHTTP	https://github.com/square/okhttp

Java UniREST	https://github.com/Kong/unirest-java
JavaScript Fetch	https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API
JavaScript jQuery	http://api.jquery.com/jquery.ajax/
JavaScript XHR	https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest
NodeJS Axios	https://github.com/axios/axios
NodeJS Http	https://nodejs.org/api/http.html
NodeJS Request	https://github.com/request/request
NodeJS UniRest	https://github.com/Kong/unirest-nodejs
Objective-C	https://developer.apple.com/library/ios/documentation/Foundation/Reference/NSURLSession_class/
OCaml	https://github.com/mirage/ocaml-cohttp
PHP CUrl	http://php.net/manual/en/ref.curl.php
PHP HttpRequest	https://www.php.net/manual/en/reserved.variables.request.php
PHP pecl_http	https://mdref.m6w6.name/http
PowerShell RestMethod	https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/invoke-restmethod?view=powershell-7
Python Built-in http.client (Python 3)	https://docs.python.org/3/library/http.client.html
Python Requests	https://requests.readthedocs.io/en/master/
Ruby Built-in NET::Http	http://docs.ruby-lang.org/en/2.0.0/Net/HTTP.html
Shell HTTPie	https://github.com/jkbrzt/httpie
Shell wget	https://www.gnu.org/software/wget/
Shell cURL	https://curl.haxx.se/
Swift Built-in NSURLSession	https://developer.apple.com/documentation/foundation/urlsession
VB.NET RestSharp	http://restsharp.org/

LOGON CODING EXAMPLES

Note: The Logon endpoint is a POST command. Any API call that uses POST would follow a similar pattern.

C – LIBCURL

```
CURL *curl;
CURLcode res;
curl = curl_easy_init();
if(curl) {
  curl_easy_setopt(curl, CURLOPT_CUSTOMREQUEST, "POST");
  curl_easy_setopt(curl, CURLOPT_URL, "https://Www.SomeServer.com/api/authenticate/logon");
  curl_easy_setopt(curl, CURLOPT_FOLLOWLOCATION, 1L);
  curl_easy_setopt(curl, CURLOPT_DEFAULT_PROTOCOL, "https");
  struct curl_slist *headers = NULL;
  headers = curl_slist_append(headers, "Content-Type: application/x-www-form-urlencoded");
  curl_easy_setopt(curl, CURLOPT_HTTPHEADER, headers);
  const char *data = "username=LAB&password=123456";
  curl_easy_setopt(curl, CURLOPT_POSTFIELDS, data);
  res = curl_easy_perform(curl);
}
curl_easy_cleanup(curl);
```

C# - RESTSHARP

```
var client = new RestClient("https://Www.SomeServer.com/api/authenticate/logon");
client.Timeout = -1;
var request = new RestRequest(Method.POST);
request.AddHeader("Content-Type", "application/x-www-form-urlencoded");
request.AddParameter("username", "LAB");
request.AddParameter("password", "123456");
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);
```

CURL

```
curl --location --request POST 'https://Www.SomeServer.com/api/authenticate/logon' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'username=LAB' \
--data-urlencode 'password=123456'
```

GO - NATIVE

```
package main

import (
  "fmt"
  "strings"
  "net/http"
  "io/ioutil"
)

func main() {

  url := "https://Www.SomeServer.com/api/authenticate/logon"
  method := "POST"

  payload := strings.NewReader("username=LAB&password=123456")

  client := &http.Client {
  }
  req, err := http.NewRequest(method, url, payload)

  if err != nil {
    fmt.Println(err)
    return
  }
  req.Header.Add("Content-Type", "application/x-www-form-urlencoded")

  res, err := client.Do(req)
  if err != nil {
    fmt.Println(err)
    return
  }
  defer res.Body.Close()

  body, err := ioutil.ReadAll(res.Body)
  if err != nil {
    fmt.Println(err)
    return
  }
  fmt.Println(string(body))
}
```

HTTP

POST /WebApiBackboneQA/api/authenticate/logon? HTTP/1.1

Host: SomeServer.com

Content-Type: application/x-www-form-urlencoded

Content-Length: 28

username=LAB&password=123456

JAVA - OKHTTP

```
OkHttpClient client = new OkHttpClient().newBuilder()
  .build();
MediaType mediaType = MediaType.parse("application/x-www-form-urlencoded");
RequestBody body = RequestBody.create(mediaType, "username=LAB&password=123456");
Request request = new Request.Builder()
  .url("https://Www.SomeServer.com/api/authenticate/logon")
  .method("POST", body)
  .addHeader("Content-Type", "application/x-www-form-urlencoded")
  .build();
Response response = client.newCall(request).execute();
```

JAVA – UNIREST

```
Unirest.setTimeouts(0, 0);
HttpResponse<String> response = Unirest.post("https://Www.SomeServer.com/api/authenticate/logon")
  .header("Content-Type", "application/x-www-form-urlencoded")
  .field("username", "LAB")
  .field("password", "123456")
  .asString();
```

JAVASCRIPT - FETCH

```
var myHeaders = new Headers();
myHeaders.append("Content-Type", "application/x-www-form-urlencoded");

var urlencoded = new URLSearchParams();
urlencoded.append("username", "LAB");
urlencoded.append("password", "123456");

var requestOptions = {
  method: 'POST',
  headers: myHeaders,
  body: urlencoded,
  redirect: 'follow'
};

fetch("https://Www.SomeServer.com/api/authenticate/logon", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

JAVASCRIPT - JQUERY

```
var settings = {
  "url": "https://Www.SomeServer.com/api/authenticate/logon",
  "method": "POST",
  "timeout": 0,
  "headers": {
    "Content-Type": "application/x-www-form-urlencoded",
      },
  "data": {
    "username": "LAB",
    "password": "123456"
  }
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

JAVASCRIPT – XHR

```
var data = "username=LAB&password=123456";

var xhr = new XMLHttpRequest();
xhr.withCredentials = true;

xhr.addEventListener("readystatechange", function() {
  if(this.readyState === 4) {
    console.log(this.responseText);
  }
});

xhr.open("POST", "https://Www.SomeServer.com/api/authenticate/logon");
xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");

xhr.send(data);
```

NODEJS – AXIOS

```
var axios = require('axios');
var qs = require('qs');
var data = qs.stringify({
 'username': 'LAB',
'password': '123456'
});
var config = {
  method: 'post',
  url: 'https://Www.SomeServer.com/api/authenticate/logon',
  headers: {
    'Content-Type': 'application/x-www-form-urlencoded',
      },
  data : data
};

axios(config)
.then(function (response) {
  console.log(JSON.stringify(response.data));
})
.catch(function (error) {
  console.log(error);
});
```

NODEJS – NATIVE

```
var https = require('follow-redirects').https;
var fs = require('fs');

var qs = require('querystring');

var options = {
  'method': 'POST',
  'hostname': 'SomeServer.com',
  'path': '/WebApiBackboneQA/api/authenticate/logon?',
  'headers': {
    'Content-Type': 'application/x-www-form-urlencoded',
      },
  'maxRedirects': 20
};

var req = https.request(options, function (res) {
  var chunks = [];

  res.on("data", function (chunk) {
    chunks.push(chunk);
  });

  res.on("end", function (chunk) {
    var body = Buffer.concat(chunks);
    console.log(body.toString());
  });

  res.on("error", function (error) {
    console.error(error);
  });
});

var postData = qs.stringify({
  'username': 'LAB',
  'password': '123456'
});

req.write(postData);

req.end();
```

NODEJS – REQUEST

```
var request = require('request');
var options = {
  'method': 'POST',
  'url': 'https://Www.SomeServer.com/api/authenticate/logon',
  'headers': {
    'Content-Type': 'application/x-www-form-urlencoded',
      },
  form: {
    'username': 'LAB',
    'password': '123456'
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
  console.log(response.body);
});
```

NODEJS – UNIREST

```
var unirest = require('unirest');
var req = unirest('POST', 'https://Www.SomeServer.com/api/authenticate/logon')
  .headers({
    'Content-Type': 'application/x-www-form-urlencoded',
      })
  .send('username=LAB')
  .send('password=123456')
  .end(function (res) {
    if (res.error) throw new Error(res.error);
    console.log(res.raw_body);
  });
```

OBJECTIVE-C NSURLSESSION

```
#import <Foundation/Foundation.h>

dispatch_semaphore_t sema = dispatch_semaphore_create(0);

NSMutableURLRequest *request = [NSMutableURLRequest requestWithURL:[NSURL URLWithString:@"http
s://Www.SomeServer.com/api/authenticate/logon"]
  cachePolicy:NSURLRequestUseProtocolCachePolicy
  timeoutInterval:10.0];
NSDictionary *headers = @{
  @"Content-Type": @"application/x-www-form-urlencoded",
  };

[request setAllHTTPHeaderFields:headers];
NSMutableData *postData = [[NSMutableData alloc] initWithData:[@"username=LAB" dataUsingEncodi
ng:NSUTF8StringEncoding]];
[postData appendData:[@"&password=123456" dataUsingEncoding:NSUTF8StringEncoding]];
[request setHTTPBody:postData];

[request setHTTPMethod:@"POST"];

NSURLSession *session = [NSURLSession sharedSession];
NSURLSessionDataTask *dataTask = [session dataTaskWithRequest:request
completionHandler:^(NSData *data, NSURLResponse *response, NSError *error) {
  if (error) {
    NSLog(@"%@", error);
  } else {
    NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *) response;
    NSError *parseError = nil;
    NSDictionary *responseDictionary = [NSJSONSerialization JSONObjectWithData:data options:0
error:&parseError];
    NSLog(@"%@",responseDictionary);
    dispatch_semaphore_signal(sema);
  }
}];
[dataTask resume];
dispatch_semaphore_wait(sema, DISPATCH_TIME_FOREVER);
```

OCAML – COHTTP

```
open Lwt
open Cohttp
open Cohttp_lwt_unix

let postData = ref "username=LAB&password=123456";;

let reqBody =
  let uri = Uri.of_string "https://Www.SomeServer.com/api/authenticate/logon" in
  let headers = Header.init ()
    |> fun h -> Header.add h "Content-Type" "application/x-www-form-urlencoded"
      in
  let body = Cohttp_lwt.Body.of_string !postData in

  Client.call ~headers ~body `POST uri >>= fun (_resp, body) ->
  body |> Cohttp_lwt.Body.to_string >|= fun body -> body

let () =
  let respBody = Lwt_main.run reqBody in
  print_endline (respBody)
```

PHP – CURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => "https://Www.SomeServer.com/api/authenticate/logon",
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => "",
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => "POST",
  CURLOPT_POSTFIELDS => "username=LAB&password=123456",
  CURLOPT_HTTPHEADER => array(
    "Content-Type: application/x-www-form-urlencoded",
      ),
));

$response = curl_exec($curl);

curl_close($curl);
echo $response;
```

PHP – HTTP_REQUEST2

```php
<?php
require_once 'HTTP/Request2.php';
$request = new HTTP_Request2();
$request->setUrl('https://Www.SomeServer.com/api/authenticate/logon');
$request->setMethod(HTTP_Request2::METHOD_POST);
$request->setConfig(array(
  'follow_redirects' => TRUE
));
$request->setHeader(array(
  'Content-Type' => 'application/x-www-form-urlencoded',
  ));
$request->addPostParameter(array(
  'username' => 'LAB',
  'password' => '123456'
));
try {
  $response = $request->send();
  if ($response->getStatus() == 200) {
    echo $response->getBody();
  }
  else {
    echo 'Unexpected HTTP status: ' . $response->getStatus() . ' ' .
    $response->getReasonPhrase();
  }
}
catch(HTTP_Request2_Exception $e) {
  echo 'Error: ' . $e->getMessage();
}
```

PHP – PECL_HTTP

```php
<?php
$client = new http\Client;
$request = new http\Client\Request;
$request->setRequestUrl('https://Www.SomeServer.com/api/authenticate/logon');
$request->setRequestMethod('POST');
$body = new http\Message\Body;
$body->append(new http\QueryString(array(
  'username' => 'LAB',
  'password' => '123456')));$request->setBody($body);
$request->setOptions(array());
$request->setHeaders(array(
  'Content-Type' => 'application/x-www-form-urlencoded',
  ));
$client->enqueue($request)->send();
```

```
$response = $client->getResponse();
echo $response->getBody();
```

POWERSHELL – RESTMETHOD

```
$headers = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$headers.Add("Content-Type", "application/x-www-form-urlencoded")

$body = "username=LAB&password=123456"

$response = Invoke-RestMethod 'https://Www.SomeServer.com/api/authenticate/logon' -
Method 'POST' -Headers $headers -Body $body
```

PYTHON – HTTP.CLIENT

```
import http.client
import mimetypes
conn = http.client.HTTPSConnection("SomeServer.com")
payload = 'username=LAB&password=123456'
headers = {
  'Content-Type': 'application/x-www-form-urlencoded',
  }
conn.request("POST", "/WebApiBackboneQA/api/authenticate/logon", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

PYTHON – REQUESTS

```
import requests

url = "https://Www.SomeServer.com/api/authenticate/logon"

payload='username=LAB&password=123456'
headers = {
  'Content-Type': 'application/x-www-form-urlencoded',
  }

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)
```

RUBY – NET::HTTP

```
require "uri"
require "net/http"

url = URI("https://Www.SomeServer.com/api/authenticate/logon")

https = Net::HTTP.new(url.host, url.port)
https.use_ssl = true

request = Net::HTTP::Post.new(url)
request["Content-Type"] = "application/x-www-form-urlencoded"
request.body = "username=LAB&password=123456"

response = https.request(request)
puts response.read_body
```

SHELL – HTTPIE

```
http --ignore-stdin --form --follow --
timeout 3600 POST https://Www.SomeServer.com/api/authenticate/logon  \
 'username'='LAB' \
 'password'='123456' \
 Content-Type:'application/x-www-form-urlencoded' \
```

SHELL – WGET

```
wget --no-check-certificate --quiet \
  --method POST \
  --timeout=0 \
  --header 'Content-Type: application/x-www-form-urlencoded' \
    --body-data 'username=LAB&password=123456' \
   'https://Www.SomeServer.com/api/authenticate/logon'
```

SWIFT - URLSESSION

```
import Foundation

var semaphore = DispatchSemaphore (value: 0)

let parameters = "username=LAB&password=123456"
let postData =  parameters.data(using: .utf8)

var request = URLRequest(url: URL(string: "https://Www.SomeServer.com/api/authenticate/logon")
!,timeoutInterval: Double.infinity)
request.addValue("application/x-www-form-urlencoded", forHTTPHeaderField: "Content-Type")

request.httpMethod = "POST"
request.httpBody = postData

let task = URLSession.shared.dataTask(with: request) { data, response, error in
  guard let data = data else {
    print(String(describing: error))
    semaphore.signal()
    return
  }
  print(String(data: data, encoding: .utf8)!)
  semaphore.signal()
}

task.resume()
semaphore.wait()
```

VB.NET - RESTSHARP

```
Dim client = New RestClient("https://Www.SomeServer.com/api/authenticate/logon")
client.Timeout = -1
Dim request = New RestRequest(Method.POST)
request.AddHeader("Content-Type", "application/x-www-form-urlencoded")
request.AddParameter("username", "LAB")
request.AddParameter("password", "123456")
Dim response As IRestResponse = client.Execute(request)

Console.WriteLine(response.Content)
```

LOGOUT CODING EXAMPLES

Note: The Logout endpoint is a GET command. Any API call that uses GET would follow a similar pattern.

C - LIBCURL

```
CURL *curl;
CURLcode res;
curl = curl_easy_init();
if(curl) {
  curl_easy_setopt(curl, CURLOPT_CUSTOMREQUEST, "GET");
  curl_easy_setopt(curl, CURLOPT_URL, "https://Www.SomeServer.com/api/authenticate/logout");
  curl_easy_setopt(curl, CURLOPT_FOLLOWLOCATION, 1L);
  curl_easy_setopt(curl, CURLOPT_DEFAULT_PROTOCOL, "https");
  struct curl_slist *headers = NULL;
  headers = curl_slist_append(headers, "Cookie: session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900");
  curl_easy_setopt(curl, CURLOPT_HTTPHEADER, headers);
  res = curl_easy_perform(curl);
}
```

```
curl_easy_cleanup(curl);
```

C# - RESTSHARP

```
var client = new RestClient("https://Www.SomeServer.com/api/authenticate/logout");
client.Timeout = -1;
var request = new RestRequest(Method.GET);
request.AddHeader("Cookie", "session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900");
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);
```

CURL

```
curl --location --request GET 'https://Www.SomeServer.com/api/authenticate/logout' \
--header 'Cookie: session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900'
```

GO – NATIVE

```
package main

import (
  "fmt"
  "net/http"
  "io/ioutil"
)

func main() {

  url := "https://Www.SomeServer.com/api/authenticate/logout"
  method := "GET"

  client := &http.Client {
  }
  req, err := http.NewRequest(method, url, nil)

  if err != nil {
    fmt.Println(err)
    return
  }
  req.Header.Add("Cookie", "session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900")
  res, err := client.Do(req)
  if err != nil {
    fmt.Println(err)
    return
  }
  defer res.Body.Close()

  body, err := ioutil.ReadAll(res.Body)
  if err != nil {
    fmt.Println(err)
    return
  }
  fmt.Println(string(body))
}
```

HTTP

GET /WebApiBackboneQA/api/authenticate/logout HTTP/1.1

Host: SomeServer.com

Cookie: session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900

JAVA – OKHTTP

```
OkHttpClient client = new OkHttpClient().newBuilder()
  .build();
Request request = new Request.Builder()
  .url("https://Www.SomeServer.com/api/authenticate/logout")
  .method("GET", null)
  .addHeader("Cookie", "session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900")
  .build();
Response response = client.newCall(request).execute();
```

JAVA – UNIREST

```
Unirest.setTimeouts(0, 0);
HttpResponse<String> response = Unirest.get("https://Www.SomeServer.com/api/authenticate/logout")
```

```
  .header("Cookie", "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900")
  .asString();
```

JAVASCRIPT –FETCH

```
var myHeaders = new Headers();
myHeaders.append("Cookie", "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900");

var requestOptions = {
  method: 'GET',
  headers: myHeaders,
  redirect: 'follow'
};

fetch("https://Www.SomeServer.com/api/authenticate/logout", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

JAVASCRIPT – JQUERY

```
var settings = {
  "url": "https://Www.SomeServer.com/api/authenticate/logout",
  "method": "GET",
  "timeout": 0,
  "headers": {
    "Cookie": "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900"
  },
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

JAVASCRIPT – XHR

```
var xhr = new XMLHttpRequest();
xhr.withCredentials = true;
xhr.addEventListener("readystatechange", function() {
  if(this.readyState === 4) {
    console.log(this.responseText);
  }
});

xhr.open("GET", "https://Www.SomeServer.com/api/authenticate/logout");
xhr.setRequestHeader("Cookie", "session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900");

xhr.send();
```

NODEJS – AXIOS

```
var axios = require('axios');

var config = {
  method: 'get',
  url: 'https://Www.SomeServer.com/api/authenticate/logout',
  headers: {
    'Cookie': 'session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900'
  }
};

axios(config)
.then(function (response) {
  console.log(JSON.stringify(response.data));
})
.catch(function (error) {
  console.log(error);
});
```

NODEJS – NATIVE

```
var https = require('follow-redirects').https;
var fs = require('fs');

var options = {
  'method': 'GET',
  'hostname': 'SomeServer.com',
  'path': '/WebApiBackboneQA/api/authenticate/logout',
  'headers': {
    'Cookie': 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
  },
  'maxRedirects': 20
};

var req = https.request(options, function (res) {
  var chunks = [];

  res.on("data", function (chunk) {
    chunks.push(chunk);
  });

  res.on("end", function (chunk) {
    var body = Buffer.concat(chunks);
    console.log(body.toString());
  });

  res.on("error", function (error) {
    console.error(error);
  });
});

req.end();
```

NODEJS – REQUEST

```
var request = require('request');
var options = {
  'method': 'GET',
  'url': 'https://Www.SomeServer.com/api/authenticate/logout',
  'headers': {
    'Cookie': 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
  }
};
request(options, function (error, response) {
  if (error) throw new Error(error);
  console.log(response.body);
});
```

NODEJS – UNIREST

```
var unirest = require('unirest');
var req = unirest('GET', 'https://Www.SomeServer.com/api/authenticate/logout')
  .headers({
    'Cookie': 'session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900'
  })
  .end(function (res) {
    if (res.error) throw new Error(res.error);
    console.log(res.raw_body);
  });
```

OBJECTIVE-C – NSURLSESSION

```
#import <Foundation/Foundation.h>

dispatch_semaphore_t sema = dispatch_semaphore_create(0);

NSMutableURLRequest *request = [NSMutableURLRequest requestWithURL:[NSURL URLWithString:@"https://Www.SomeServer.com/api/authenticate/logout"]
  cachePolicy:NSURLRequestUseProtocolCachePolicy
  timeoutInterval:10.0];
NSDictionary *headers = @{
  @"Cookie": @"session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900"
};

[request setAllHTTPHeaderFields:headers];

[request setHTTPMethod:@"GET"];

NSURLSession *session = [NSURLSession sharedSession];
NSURLSessionDataTask *dataTask = [session dataTaskWithRequest:request
completionHandler:^(NSData *data, NSURLResponse *response, NSError *error) {
  if (error) {
    NSLog(@"%@", error);
  } else {
    NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *) response;
    NSError *parseError = nil;
    NSDictionary *responseDictionary = [NSJSONSerialization JSONObjectWithData:data options:0 error:&parseError];
    NSLog(@"%@",responseDictionary);
    dispatch_semaphore_signal(sema);
  }
}];
```

```
[dataTask resume];
dispatch_semaphore_wait(sema, DISPATCH_TIME_FOREVER);
```

OCAML – COHTTP

```
open Lwt
open Cohttp
open Cohttp_lwt_unix

let reqBody =
  let uri = Uri.of_string "https://Www.SomeServer.com/api/authenticate/logout" in
  let headers = Header.init ()
    |> fun h -> Header.add h "Cookie" "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900"
  in
  Client.call ~headers `GET uri >>= fun (_resp, body) ->
  body |> Cohttp_lwt.Body.to_string >|= fun body -> body

let () =
  let respBody = Lwt_main.run reqBody in
  print_endline (respBody)
```

PHP – CURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => "https://Www.SomeServer.com/api/authenticate/logout",
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => "",
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => "GET",
  CURLOPT_HTTPHEADER => array(
    "Cookie: session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900"
  ),
));

$response = curl_exec($curl);
```

```
curl_close($curl);
echo $response;
```

PHP – HTTP_REQUEST2

```
<?php
require_once 'HTTP/Request2.php';
$request = new HTTP_Request2();
$request->setUrl('https://Www.SomeServer.com/api/authenticate/logout');
$request->setMethod(HTTP_Request2::METHOD_GET);
$request->setConfig(array(
  'follow_redirects' => TRUE
));
$request->setHeader(array(
  'Cookie' => 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
));
try {
  $response = $request->send();
  if ($response->getStatus() == 200) {
    echo $response->getBody();
  }
  else {
    echo 'Unexpected HTTP status: ' . $response->getStatus() . ' ' .
    $response->getReasonPhrase();
  }
}
catch(HTTP_Request2_Exception $e) {
  echo 'Error: ' . $e->getMessage();
}
```

PHP – PECL_HTTP

```
<?php
$client = new http\Client;
$request = new http\Client\Request;
$request->setRequestUrl('https://Www.SomeServer.com/api/authenticate/logout');
$request->setRequestMethod('GET');
$request->setOptions(array());
$request->setHeaders(array(
  'Cookie' => 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
));
```

```
$client->enqueue($request)->send();
$response = $client->getResponse();
echo $response->getBody();
```

POWERSHELL – RESTMETHOD

```
$headers = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$headers.Add("Cookie", "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900")

$response = Invoke-RestMethod 'https://Www.SomeServer.com/api/authenticate/logout' -
Method 'GET' -Headers $headers -Body $body
```

PYTHON – HTTP.CLIENT

```
import http.client
import mimetypes
conn = http.client.HTTPSConnection("SomeServer.com")
payload = ''
headers = {
  'Cookie': 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
}
conn.request("GET", "/WebApiBackboneQA/api/authenticate/logout", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

PYTHON – REQUESTS

```
import requests

url = "https://Www.SomeServer.com/api/authenticate/logout"

payload={}
headers = {
  'Cookie': 'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.text)
```

RUBY – NET::HTTP

```
require "uri"
require "net/http"

url = URI("https://Www.SomeServer.com/api/authenticate/logout")

https = Net::HTTP.new(url.host, url.port)
https.use_ssl = true

request = Net::HTTP::Get.new(url)
request["Cookie"] = "session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900"

response = https.request(request)
puts response.read_body
```

SHELL – HTTPIE

```
http --follow --timeout 3600 GET https://Www.SomeServer.com/api/authenticate/logout \
 Cookie:'session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900'
```

SHELL – WGET

```
wget --no-check-certificate --quiet \
  --method GET \
  --timeout=0 \
  --header 'Cookie: session=session-
id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelim
it=900' \
   'https://Www.SomeServer.com/api/authenticate/logout'
```

SWIFT – URLSESSION

```
import Foundation

var semaphore = DispatchSemaphore (value: 0)

var request = URLRequest(url: URL(string: "https://Www.SomeServer.com/api/authenticate/logout"
)!,timeoutInterval: Double.infinity)
```

```
request.addValue("session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900", forHTTPHeaderField: "Cookie")

request.httpMethod = "GET"

let task = URLSession.shared.dataTask(with: request) { data, response, error in
  guard let data = data else {
    print(String(describing: error))
    semaphore.signal()
    return
  }
  print(String(data: data, encoding: .utf8)!)
  semaphore.signal()
}

task.resume()
semaphore.wait()
```

VB.NET – RESTSHARP

```
Dim client = New RestClient("https://Www.SomeServer.com/api/authenticate/logout")
client.Timeout = -1
Dim request = New RestRequest(Method.[GET])
request.AddHeader("Cookie", "session=session-id=kUn8axg5aDzfBOuVqs2QwF2pZiaI%2fv%2fQSsQt2mFq000%3d&username=346650&epoch=1604350665&timelimit=900")
Dim response As IRestResponse = client.Execute(request)

Console.WriteLine(response.Content)
```

FOR MORE INFORMATION

Please contact Psyche Systems technical support.