Illumina NovaSeq LIMS API v1



Contents

[Introduction 3](#_Toc505705683)

[API Specification 3](#_Toc505705684)

[Login URL Retrieval 3](#_Toc505705685)

[Recipe Retrieval 4](#_Toc505705686)

[Providing Run Metrics 5](#_Toc505705687)

[Providing Run Progress 5](#_Toc505705688)

[DTOs and Structures 6](#_Toc505705689)

[RecipeRequestDTO 6](#_Toc505705690)

[RecipeRequestResponseDTO 6](#_Toc505705691)

[SequencingRunMetrics 7](#_Toc505705692)

[SequencingRunStatusDTO 9](#_Toc505705693)

[RunInfo 9](#_Toc505705694)

[Reagent 11](#_Toc505705695)

[Enumeration: SequencingRunStatus 11](#_Toc505705696)

[Enumeration: SequencingInstrumentType 12](#_Toc505705697)

[Authentication 13](#_Toc505705698)

[Login Page 13](#_Toc505705699)

# Introduction

The NovaSeq line of sequencers provide support for an external LIMS system. Illumina has developed an API that can be used by an external LIMS system to interact with a NovaSeq instrument. This document will describe this API, along with authentication details that are required to successfully interoperate with a NovaSeq instrument.

# API Specification

The NovaSeq LIMS API is a REST API that a given LIMS service must implement. NovaSeq will call the endpoints of this API at specific times during sequencing run setup and progression. All interactions with the LIMS system will be conducted through this API, with the notable exception of user login which will be covered in another section.

## Login URL Retrieval

Figure 1 - Retrieving Login URL

|  |  |
| --- | --- |
| Purpose | The LIMS service will provide a URL to the Login Server, which will be used to access the user authentication token which will be used for the remainder of the sequencing run. |
| URL | Illumina/Sequencer/v2/sequencing-run/login |
| Method | GET |
| Success Response | A string representation of the login URL, e.g. “https://localhost:9001” |

The URL returned should be a well-formed, usable URL. An example would be <https://localhost:9001>. NovaSeq Control Software will use this URL to furnish the user login page during sequencing setup.

## Recipe Retrieval

Figure 2 - LIMS Recipe Retrieval

|  |  |
| --- | --- |
| Purpose | The LIMS service will provide the information required to successfully setup a NovaSeq recipe and sequencing run |
| URL | Illumina/Sequencer/v2/sequencing-run/recipe/novaseq |
| Method | POST |
| DTO | {  RecipeRequestDTO (see DTOs and Structures below)  } |
| Success Response | {  RecipeRequestResponseDTO (see DTOs and Structures below)  } |

NovaSeq Control Software will send a DTO across with the library container ID and, in certain scenarios, the flow cell ID. LIMS will then send across a DTO populated with information needed to setup a NovaSeq sequencing run.

## Providing Run Metrics

Figure 3 - Submitting Run Metrics

|  |  |
| --- | --- |
| Purpose | NovaSeq will provide run metrics on successful completion of a sequencing run |
| URL | Illumina/Sequencer/v2/sequencing-run/metrics |
| Method | POST |
| DTO | {  SequencingRunMetrics (see DTOs and Structures below)  } |
| Success Response | HTTP Status Code |

## Providing Run Progress

Figure 4 - Submitting Run Progress

|  |  |
| --- | --- |
| Purpose | NovaSeq will provide run progress updates throughout a sequencing run |
| URL | Illumina/Sequencer/v2/sequencing-run/update |
| Method | POST |
| DTO | {  SequencingRunStatusDTO (see DTOs and Structures below)  } |
| Success Response | HTTP Status Code |

# DTOs and Structures

The following DTOs and structures are sent by NovaSeq Control Software to the LIMS service throughout a sequencing run.

## RecipeRequestDTO

Figure 5 - RecipeRequestDTO

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| FlowCellId | string | A string representation of the flow cell barcode, e.g. H7GWDMCVY |
| LibraryContainerId | string | A string representation of the library tube barcode, e.g. NV0001375-LIB |

## RecipeRequestResponseDTO

Figure 6 - RecipeRequestResponseDTO

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| run\_name | string | A non-whitespace string consisting of on letters, digits, ‘\_’ and ‘-‘ |
| run\_mode | string | A string representation of the following values: |
| workflow\_type | string | A string representation of the following values:   * NoIndex, * SingleIndex, * DualIndex |
| librarytube\_id | string | A string representation of the library tube barcode, e.g. NV0001375-LIB |
| flowcell\_id | string | A string representation of the flow cell barcode, e.g. H7GWDMCVY |
| sample\_loading\_type | string | A string representation of the following values:   * NovaSeqStandard, * NovaSeqXp |
| rehyb | bool | A Boolean that is true if the run is being rehybed (Note: rehyb is not currently supported – this field should be set to ‘false’) |
| paired\_end | bool | A Boolean that is true if the run is a paired end run |
| read1 | int | The number of cycles in read 1. The number cannot be:   * Less than 2 * The total of all read cycles cannot be greater than the amount of cycles supported by the reagent kit |
| read2 | int | The number of cycles in read 2. The number cannot be:   * Equal to 1 * The total of all read cycles cannot be greater than the amount of cycles supported by the reagent kit |
| index\_read1 | int | The number of cycles in index read 1. The number cannot be:   * Equal to 1 * Greater than 20 * The total of all read cycles cannot be greater than the amount of cycles supported by the reagent kit |
| index\_read2 | int | The number of cycles in index read 1. The number cannot be:   * Equal to 1 * Greater than 20 * The total of all read cycles cannot be greater than the amount of cycles supported by the reagent kit |
| output\_folder | string | A string representation of the run output folder on the local NovaSeq system |
| samplesheet | string | A string representation of the name of the sample sheet file |
| usecustomrecipe | bool | A Boolean that is true if the run is using a custom recipe |
| customRecipe | string | A string representation of the path to the custom recipe |
| use\_basespace | bool | A Boolean that is true if the run is using BaseSpace |
| basespace\_mode | string | A string representation of the following values:   * RunMonitoringAndStorage * RunMonitoringOnly |
| use\_custom\_read1\_primer | bool | A Boolean that is true if the run is using a custom read 1 primer |
| use\_custom\_read2\_primer | bool | A Boolean that is true if the run is using a custom read 2 primer |
| use\_custom\_index\_read1\_primer | bool | A Boolean that is true if the run is using a custom index read 1 primer |

## SequencingRunMetrics

Figure 7 - SequencingRunMetrics

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| RunInfo | RunInfo | A RunInfo object, defined below |
| Status | SequencingRunStatus | A SequencingRunStatus value, defined below |
| YieldPfR1 | float | A float value representing the amount of the read 1 yield passing filter |
| YieldPfR2 | float | A float value representing the amount of the read 2 yield passing filter |
| ReadsPfR1 | float | A float value representing the number of read 1 clusters passing filter |
| ReadsPfR2 | float | A float value representing the number of read 2 clusters passing filter |
| ClusterDensityR1 | float | A float value representing the read 1 cluster density |
| ClusterDensityR2 | float | A float value representing the read 2 cluster density |
| PercentPfR1 | float | A float value representing the read 1 percent passing filter |
| PercentPfR2 | float | A float value representing the read 2 percent passing filter |
| PercentGreaterThanQ30R1 | float | A float value representing the read 1 percent greater than Q30 |
| PercentGreaterThanQ30R2 | float | A float value representing the read 2 percent greater than Q30 |
| IntensityCycle1R1 | float | A float value representing the read 1 cycle 1 intensity |
| IntensityCycle1R2 | float | A float value representing the read 1 cycle 2 intensity |
| PercentAlignedR1 | float | A float value representing the read 1 percent aligned |
| PercentAlignedR2 | float | A float value representing the read 2 percent aligned |
| PercentErrorRateR1 | float | A float value representing the read 1 percent error rate |
| PercentErrorRateR2 | float | A float value representing the read 2 percent error rate |
| PercentPhasingR1 | float | A float value representing the read 1 percent phasing |
| PercentPhasingR2 | float | A float value representing the read 2 percent phasing |
| PercentPrePhasingR1 | float | A float value representing the read 1 percent prephasing |
| PercentPrePhasingR2 | float | A float value representing the read 2 percent prephasing |

## SequencingRunStatusDTO

Figure 8 - SequencingRunStatusDTO

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| RunInfo | RunInfo | A RunInfo object, defined below |
| Status | SequencingRunStatus | A SequencingRunStatus value, defined below |
| Reagents | Reagent[] | An array of Reagent objects, defined below |
| CurrentCycle | int | The currently executing cycle |
| CurrentRead | int | The currently executing read |
| InstrumentControlSoftwareVersion | string | The version of the control software |
| RtaVersion | string | The version of RTA |
| FirmwareVersion | string | The version of Firmware |

## RunInfo

Figure 9 - RunInfo

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| RunId | string | A string that follows a configurable template, defaulted to “{YY}{MM}{DD}\_{SERIAL}\_{RUN}\_{FCPOS}{FCSERIAL}”,  where:   * YY = year * MM = month * DD = day * Serial = instrument name * Run = total number of sequencing runs on the instrument * FCPOS = flow cell side, ‘A’ or ‘B’ * FCSERIAL = flow cell barcode |
| FlowCellId | string | A string representation of the flow cell barcode, e.g. H7GWDMCVY |
| LibraryTubeId | string | A string representation of the library tube barcode, e.g. NV0001375-LIB |
| InstrumentId | string | The name of the instrument |
| InstrumentType | SequencingInstrumentType | A value from the SequencingInstrumentType enum, defined below |
| FlowCellSide | string | The flow cell side, ‘Left’ or ‘Right’ |
| DateTime | DateTime | The time the run info was created, formatted as “2/6/2018 6:31:49 PM” |
| OutputFolder | string | A string representation of the run output folder on the local NovaSeq system |
| UserName | string | The user name of the LIMS logged in user |

## Reagent

Figure 10 - Reagent

|  |  |  |
| --- | --- | --- |
| **NAME** | **TYPE** | **INFO** |
| Name | string | A string representation of the following values:   * Flow Cell, * SBS, * Library Tube, * Cluster, * Buffer |
| ExpirationDate | DateTime | The expiration date of the reagent kit |
| LotNumber | string | The lot number of the reagent kit |
| SerialNumber | string | The serial number of the reagent kit |
| PartNumber | string | The part number of the reagent kit |
| Mode | string | A string representation of the following values:   * S2, * S1, * HTWashOnly , * LTWashOnly, * S4 |
| Cycles | int | The supported cycles of the reagent kit |

In addition to the above, two enumerations are defined for these communications.

## Enumeration: SequencingRunStatus

Figure 11 - SequencingRunStatus Enumeration

SequencingRunStatus

{

RunStarted,

RunEndedByUser,

RunErroredOut,

RunCompletedSuccessfully

}

## Enumeration: SequencingInstrumentType

Figure 12- SequencingInstrumentState Enumeration

SequencingInstrumentType

{

NovaSeq6000

}

# Authentication

NovaSeq utilizes OAuth 2 bearer tokens for authentication. These tokens must be provided by the LIMS furnished login page, which will be displayed in NovaSeq Control Software.

# Login Page

When requested, the LIMS service will provide to NovaSeq Control Software a URL for a login page. NovaSeq Control Software will not store any usernames or passwords used in the login page, it will simply utilize the token provided for authentication.

As with most LIMS situations, there is a need for user tracking. Since NovaSeq will not be saving any user provided login information, the login server needs to return, in addition to the token, the username. At certain points in the run (See DTOs and Structures above), this username will be provided to the LIMS service. The username should be added to the JSON as a field entitled “UserName”. An example JSON can be seen below.

Figure 13 - Sample Login JSON

{

"access\_token": "sampleToken",

"token\_type": "bearer",

"expires\_in": 172799,

"UserName": "user",

".issued": "Tue, 06 Feb 2018 19:47:55 GMT",

".expires": "Thu, 08 Feb 2018 19:47:55 GMT"

}

NovaSeq Control Software will interact with the login page through the JavaScript structure ‘window.external’. The username and token will need to be set there. The JavaScript will need to call three methods on “window.external”:

* + SetUserName(string)
  + SetToken(string)
  + LoginComplete()

Sample JavaScript code to set those fields can be seen below.

Figure 14 - JavaScript Code Setting Necessary Values

window.sessionStorage.accessToken = json.access\_token;

window.sessionStorage.setItem("username", json.UserName);

window.external.SetUserName(window.sessionStorage.username);

window.external.SetToken(window.sessionStorage.accessToken);

window.external.LoginComplete();