master document for generating PDF

= GOBii API

Table of Contents

Overview

The GOBii web service API provides a comprehensive set of RESTful methods for accomplishing the following core tasks in the GOBii Genomic Data Management (GDM) system. This manual explains the GOBii web architecture and its web service operations.

Version information

Version : 1.0

License information

License : Apache 2.0
License URL : http://www.apache.org/licenses/LICENSE-2.0.html

URI scheme

Host : localhost:8081
BasePath : /api
Schemes : HTTP

System Architecture

DTO Models

Content TBD

Data Access

Content TBD

DTO Maping

Content TBD

Services

Content TBD

The Web Controller

Content TBD

Web Service Operations

Request/Response Structure

The http body structure is the same for all responses as well as for the body of POST and PUT requests. Here is a high level schematic of the body structure:

```
{
    "payload":{
        "linkCollection":{
            "linksPerDataItem":[

            ],
            "exploreLinksPerDataItem":[

            ]
        },
        "data":[

        ]
    },
    "header":{
        "gobiiProcessType":"",
        "dtoHeaderAuth":{
            "userName":null,
            "password":null,
            "token":"",
            "gobiiCropType":""
        },
        "status":{
            "succeeded":true,
            "statusMessages":[

            ],
            "statusMessagesByCode":{

            }
        },
        "pagination":null,
        "gobiiVersion":"2.0_2018-07-31",
        "cropType":""
    }
}
```

For requests, the following keys can be omitted:

- payload.linkCollection
- header

The content of the "data" array will vary depending on the type of the call. For example, for a POST to the the /project resource, it will contain a project object.

After a successful POST to the /auth resource, the "token" key in "dtoHeaderAuth" contains the token value, which should be used to set the X-Auth-Token header for subsequent requests.

Authentication

A valid token may be acquired by doing a POST to the /auth resource with an empty body and the following request headers set:

X-Username X-Password

The username and password depend on how the destination GOBii server is configured (per the gobii-web.xml file):

If gobiiAuthenticationType of the server is TEST, the username and password are hard-coded to USER_READER and reader, respectively. If the server is configured for LDAP authentication: The username/password must be provisioned in the configured LDAP server (if it is not, the error from the server will be 401: Unauthorized) There must be an entry for the user in contact table, where the username value matches the ldap user name; NOTE: if no such record is found, the error will be 403: Forbidden: Missing contact info for user <user-name> in crop database <crop-db>; a contact record must have username = <user-name>

When authentication succeeds, the response headers will contain a X-Auth-Token key whose value is a valid token. For subsequent requests, it is sufficient to supply X-Auth-Token in the request headers with the token that was received from the /auth POST. Some browsers prevent the client code from reading custom response headers. In this case, the token can also be retrieved from the meta data portion of the response body (see below).

Error Handling

Errors internal to GOBii are reported in the "header" key accompanying the response. An example of how to retrieve error data is provided in the following code fragment.

```
if (!header.getStatus().isSucceeded() ||
        header
                .getStatus()
                .getStatusMessages()
                .stream()
                .filter(headerStatusMessage ->
headerStatusMessage.getGobiiStatusLevel().equals(GobiiStatusLevel.VALIDATION))
                .count() > 0) {
    returnVal = true;
    System.out.println();
    System.out.println("*** Header errors: ");
    for (HeaderStatusMessage currentStatusMesage :
header.getStatus().getStatusMessages()) {
        System.out.println(currentStatusMesage.getMessage());
    }
}
```

Documentation Tools

The toolchain used to produce this documentation is swagger compatible. It is

Paths

/analyses

```
POST /gobii/v1/analyses
```

Description

Creates an analysis entity. Consult the Request/Response Section before attempting to use this call.

Parameters

Type	Name	Description	Schema	Default
Body	body		PayloadEnvelopeAnalysisDTO	

Responses

HTTP Code	Description	Schema
200	successful operation	PayloadEnvelopeAnalysisDTO

/auth

```
POST /gobii/v1/auth
```

Description

The user credentials are specified in the request headers; the response and the response headers incldue the token, which can which the client will include in the request headers for subsequent requests.

Parameters

Type	Name	Description	Schema	Default
Body	body		string	

Responses

HTTP Code	Description	Schema
200	successful operation	string

/configsettings

```
GET /gobii/v1/configsettings
```

Description

Provides generic configuration information about the GOBii instance

Responses

HTTP Code	Description	Schema
200	successful operation	[PayloadEnvelopeConfigSettingsDTO](#)

Definitions

AnalysisDTO

Name	Description	Schema
algorithm		string
allowedProcesssTypes		< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
analysisDescription		string
analysisId		integer(int32)
analysisName		string
anlaysisTypeId		integer(int32)
createdBy		integer(int32)
createdDate		string(date-time)
entityNameType		enum (UNKNOWN, ANALYSIS, CONTACT, DATASET, CV, CVGROUP, PROJECT, ORGANIZATION, PLATFORM, MANIFEST, MAPSET, MARKER_GROUP, EXPERIMENT, REFERENCE, ROLE, DISPLAY, MARKER, PROTOCOL, VENDOR_PROTOCOL, DNASAMPLE, LINKAGE_GROUP, DNARUN, GERMPLASM, JOB)
id		integer(int32)
modifiedBy		integer(int32)

Name	Description	Schema
modifiedDate		string(date-time)
parameters		< EntityPropertyDTO > array
program		string
programVersion		string
referenceId		integer(int32)
sourceName		string
sourceUri		string
sourceVersion		string
statusId		integer(int32)
timeExecuted		string(date-time)

ConfigSettingsDTO

Name	Description	Schema
allowedProcessTypes		< enum (CREATE, READ, UPDATE, DELETE, NONE) > array
id		integer(int32)
maxUploadSizeMbytes		integer(int32)
serverCapabilities		< string, boolean > map
serverConfigs		< string, ServerConfigItem > map

EntityPropertyDTO

Name	Description	Schema
entityIdId		integer(int32)
propertyId		integer(int32)
propertyName		string
propertyValue		string

Header

Name	Description	Schema
cropType		string
dtoHeaderAuth		HeaderAuth
gobiiProcessType		enum (CREATE, READ, UPDATE, DELETE, NONE)
gobiiVersion		string
maxGet		integer(int32)
maxPost		integer(int32)
maxPut		integer(int32)
pagination		Pagination
status		Status

HeaderAuth

Name	Description	Schema
gobiiCropType		string
password		string
token		string
userName		string

HeaderStatusMessage

Name	Description	Schema
gobiiStatusLevel		enum (ERROR, VALIDATION, WARNING, INFO, OK)

Name	Description	Schema
gobiiValidationStatusType		enum (NONE, UNKNOWN, VALIDATION_COMPOUND_UNIQUE, VALIDATION_NOT_UNIQUE, NONEXISTENT_FK_ENTITY, BAD_REQUEST, MISSING_REQUIRED_VALUE, ENTITY_DOES_NOT_EXIST, ENTITY_ALREADY_EXISTS, UNKNOWN_ENUM_VALUE, RESOURCE_LIMIT)
message		string

Link

Name	Description	Schema
description		string
href		string
methods		< enum (GET, POST, PUT, PATCH, DELETE, OPTIONS) > array

LinkCollection

Name	Description	Schema
exploreLinksPerDataItem		< < Link > array > array
linksPerDataItem		< Link > array

Pagination

Name	Description	Schema
currentPage		integer(int32)
pageSize		integer(int32)
pagedQueryId		string

Name	Description	Schema
queryTime		string(date-time)
totalPages		integer(int32)

PayloadAnalysisDTO

Name	Description	Schema
data		< AnalysisDTO > array
linkCollection		LinkCollection

PayloadConfigSettingsDTO

Name	Description	Schema
data		< ConfigSettingsDTO > array
linkCollection		LinkCollection

PayloadEnvelopeAnalysisDTO

Name	Description	Schema
header		Header
payload		PayloadAnalysisDTO

PayloadEnvelopeConfigSettingsDTO

Name	Description	Schema
header		Header
payload		PayloadConfigSettingsDTO

ServerConfigItem

Name	Description	Schema
confidentialityNotice		string
contextRoot		string
domain		string
fileLocations		< string, string > map
gobiiCropType		string

Name	Description	Schema
port		integer(int32)

Status

Name	Description	Schema
statusMessages		< HeaderStatusMessage > array
statusMessagesByCode		< string, string > map
succeeded	**Default** : `false`	boolean