External API V2

Breda, Thursday, December 10, 20

HK

Version: v0.1

Status: CONCEPT

**Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | **Change** | **Date** | **Version** |
| H.C. Korteweg | First version of the document | 10/12/2020 | V0.1 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1. Goal/Objective of this document 3](#_Toc58515969)

[2. Roles and responsibilities 3](#_Toc58515970)

[3. Current functionality version 1. 3](#_Toc58515971)

[4. New functionality in version 2. 3](#_Toc58515972)

[5. Design, UI and Wireframes 5](#_Toc58515973)

## Goal/Objective of this document

CYS’s APIs are Restful. They allow you to perform 'RESTful' operations such as reading, modifying, adding or deleting data in imports, projects, and schedulers in CYS.

The current functionality consists of:

Importing json formats, XLSX and csv to an importslot

Exporting json format, XLSX and csv

Dashboard (aggregated data) json format,

## Roles and responsibilities

|  |  |  |
| --- | --- | --- |
| Name | Role | Responsibility |
| Taco van der Pompe | Project Manager | Supports decision-making process and reflects on content and process.  Leads the process. |
|  |  |  |

## Current functionality version 1.

Importing data in an importslot json format or csv or xlsx file. Json import will be released soon:

Exporting data from an importslot.

Viewing data from a project

Exporting data from a project

Viewing results from a dashboard.

The current API’s are:

1. Get public/api/project
2. Get public/api/project/exportdata
3. Get public/api/import
4. Post public/api/import
5. Post public/api/importjson
6. Get public/api/import/exportdata
7. Get public/api/dashboard
8. Post public/api/dashboard

## New functionality in version 2.

In the new version of the external APIs we should add several filter options parameters and methods.

**Add APIs**

Export Images

This could be a nice feature if a customer wants to export the recorded videos or images from a project.

Response Export

This could be a nice feature if a customer wants to export the aggregated results.

Blacklist API

This could be a nice feature if a customer wants to import blacklisted records from other systems they use, otherwise they need to do this by hand or import.

Scheduler API

The scheduler API is not available right now but should be built in the current version. So the Scheduler API isn’t part of this document.

**Add methods:**

PATCH

Used for updating or modifying an entry in a project, PATCH is going to be similarly limited like PUT, though in some implementations it might be used to modify specific entries that are unique to the user.

DELETE

Delete is specifically used to delete an entry from the project, and as such, is often heavily limited in its use and requires higher rights to utilize.

**Add headers:**

Add Overload headers

X-RateLimit-Limit Maximum number of requests per minute.

X-RateLimit-Remaining Number of requests remaining in this period.

Add Pagination header

For performance reasons the export APIs should paginates the response output. This is because returning the entire data set might be feasible for some queries but prohibitive for others that return a very large amount of data.

**Add parameters**

Add Selecting columns parameter

It should be possible to select or deselect columns in the project and export API’s. For this we need to add a parameter For example columns.select() or columns.deselect()

Add Range selection (possible available right now)

It should be possible to add a range when a filter is used.

for example sys\_completedDate is after mm/dd/yyyy and is before mm/dd/yyyy

Add Updating records parameter

When a rule in the datagarden is executed a special column should be flagged as “True” this column should be named for example sys\_changed. A filter can be used to select only the changed values for example sys\_changed = True.

If the parameter sys\_changed = True is used. The sys\_changed value should be set to “false” after calling the requested record

## Design, UI and Wireframes

The examples will be created based on the functions which will be added