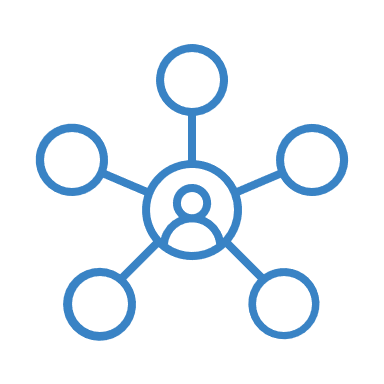


CONTACTS

 info@uppwise.com

 [LinkedIn Company Profile](https://www.linkedin.com/company/uppwise)

 [www.uppwise.com](http://www.uppwise.com)

API Guide

Reporting Extract Data

Initiative Module

Table of Contents

[1. Scope Of The Document 3](#_Toc128491536)

[2. Versioning of the API 3](#_Toc128491537)

[3. API Parameters 3](#_Toc128491538)

[4. Default API Parameters Table 3](#_Toc128491539)

[5. Authentication 4](#_Toc128491540)

[5.1. Input Parameters 4](#_Toc128491541)

[5.2. Output 5](#_Toc128491542)

[5.3. Token Generation from the Application 6](#_Toc128491543)

[6. How to connect to excel 7](#_Toc128491544)

[7. How to connect to power BI tool 10](#_Toc128491545)

[8. InitiativeRegisterSQLDSView 13](#_Toc128491546)

[8.1. Input Parameters 15](#_Toc128491547)

[8.2. Output 16](#_Toc128491548)

[9. PROJECTRegisterEPICSQLDS 17](#_Toc128491549)

[9.1. Input Parameters 19](#_Toc128491550)

[9.2. Output 20](#_Toc128491551)

[10. PROJECTRegisterPROJECTSQLDS 22](#_Toc128491552)

[10.1. Input Parameters 24](#_Toc128491553)

[10.2. Output 24](#_Toc128491554)

[11. InitiativeMapRegisterSQLDs 26](#_Toc128491555)

[11.1. Input Parameters 27](#_Toc128491556)

[11.2. Output 28](#_Toc128491557)

[12. ResourceAvailabilitySQLDS 29](#_Toc128491558)

[12.1. Input Parameters 31](#_Toc128491559)

[12.2. Output 31](#_Toc128491560)

[13. ResourceMapSQLDS 32](#_Toc128491561)

[13.1. Input Parameters 34](#_Toc128491562)

[13.2. Output 34](#_Toc128491563)

[14. InitiativeEffortDetailsbyPeriod 35](#_Toc128491564)

[14.1. Input Parameters 38](#_Toc128491565)

[14.2. Output 39](#_Toc128491566)

[15. FeatureSQLDS 41](#_Toc128491567)

[15.1. Input Parameters 43](#_Toc128491568)

[15.2. Output 43](#_Toc128491569)

[16. ProjectCOSAssociationSQLDS 44](#_Toc128491570)

[16.1. Input Parameters 45](#_Toc128491571)

[16.2. Output 46](#_Toc128491572)

[17. ScheduleDataSQLDS 47](#_Toc128491573)

[17.1. Input Parameters 48](#_Toc128491574)

[17.2. Output 49](#_Toc128491575)

[18. InitiativeSummary 50](#_Toc128491576)

[18.1. Input Parameters 53](#_Toc128491577)

[18.2. Output 53](#_Toc128491578)

# Scope Of The Document

Exposing of the Reporting Data from the SPM Analytics API These API endpoints is currently consumed over the Dashboard application the same can be used to retrieve data externally,

User: supplied separately

Pwd: supplied separately

|  |  |  |
| --- | --- | --- |
| Environment | | BaseUrl |
| **Production** | [https://spm.uppwise.com/ReportingAPI\_SPMAnalytics/api](file:///C:/Users/LENOVO/Downloads/%20https:/spm.uppwise.com/ReportingAPI_SPMAnalytics/api) | |

# Versioning of the API

Some of the API has versioning implemented the purpose of the versioning was to not impact the existing layout created over the SPM Analytics Application. In this document we are considering the latest API that are available.

# API Parameters

To get the proper data from the API end points there is a need to pass the correct parameters to the API, there are few parameters that are necessary for the proper fetch of data from the API,

The API required below parameters

* **Token** – For the authentication to get the exposed data user with the Global Administrator rights.

# Default API Parameters Table

Please consider this table for all the Parameters API Endpoints.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Required to generate Please refer to the Authentication below |

# Authentication

For all the data that is exposed through the SPM Analytics API for which a token is required, the token is generated through another API which is a POST request the token is valid as long as there is no change of the user credential if there is a change in user credential there is a need to generate a new token **api/v1/Authentication/GetJwtToken**

Below are the steps to generate a token for the SPM Analytics API

|  |  |
| --- | --- |
| **HTTP Method** | POST |
| **Endpoint** | **<** Base Url **>/v1/Authentication/GetJwtToken** |
| **HTTP Headers** | none |

## Input Parameters

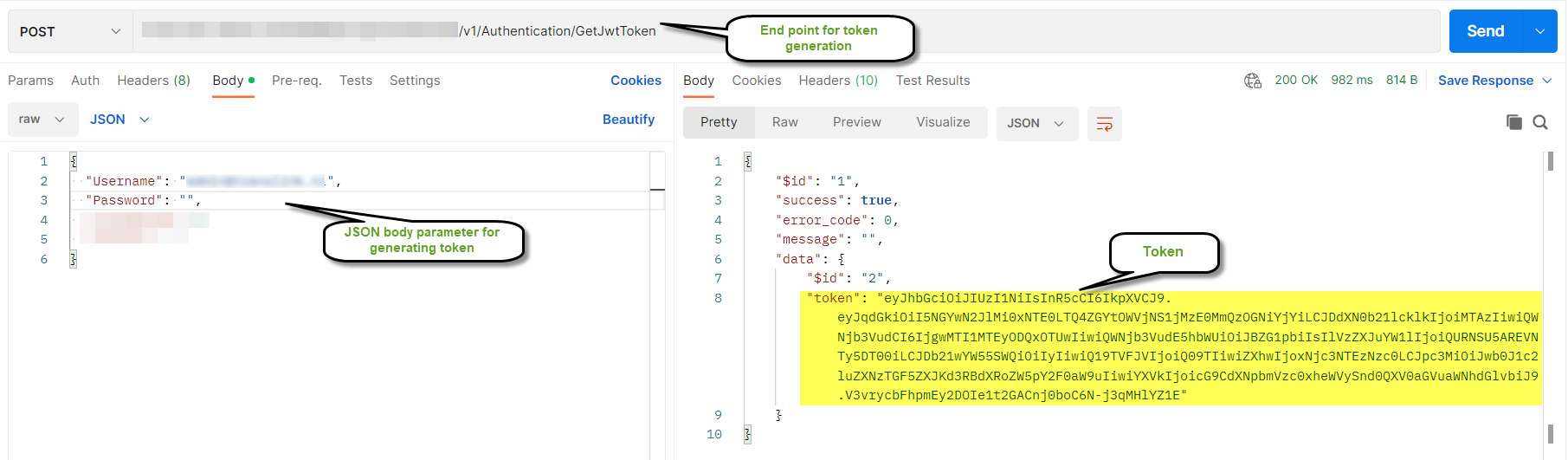
The method accepts a two Json body Parameter for the authentication

* username– the user with the Global Administrator rights
* password- the user password credentials

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {username} | the user with the Global Administrator rights | String | Yes | Please consider the Default API Parameters Table |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| {password} | The user password credentials | String | Yes | Please consider the Default API Parameters Table |

* To generate a token the user credential is passed as the body parameter is in Json form.
* There are two Json property i.e. username and password
* The endpoint for the token generation is **<** BaseUrl **>/v1** **/Authentication/GetJwtToken**



## Output

On successfully authentication structure of the output JSON document returned from the API in which token key is present.

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": {

"$id": "2",

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJqdGkiOiIxZGE0ZjNiZC0yZTU3LTQ1M2ItYmY5NS05NTA5NzRhNWE1NWEiLCJDdXN0b21lcklkIjoiMCIsIkFjY291bnQiOiI4MDEyNTExMjg0MTk1MCIsIkFjY291bnROYW1lIjoiQWRtaW4iLCJVc2VybmFtZSI6IkFETUlOQFRSQU5TTElOSy5OTCIsIkNvbXBhbnlJZCI6IjIiLCJDX1NUUlUiOiJDT1MiLCJleHAiOjE2NzIzOTQ0MzUsImlzcyI6InBvQnVzaW5lc3NMYXllckp3dEF1dGhlbmljYXRpb24iLCJhdWQiOiJwb0J1c2luZXNzTGF5ZXJKd3RBdXRoZW5pY2F0aW9uIn0.anCHTRmVj144V0X5xuWuafWAUaV9\_eaGJXNl\_qVRfpI"

}

In case of any error during the authentication, the following JSON is returned

{

"$id": "1",

"success": **false**,

"error\_code": 2,

"message": "Invalid Credentials",

"data": **null**

}

## Token Generation from the Application

In SPM Application in on clicking on the user image My Preferences>General>SPM Integration the token is coming which will used in authenticaation.

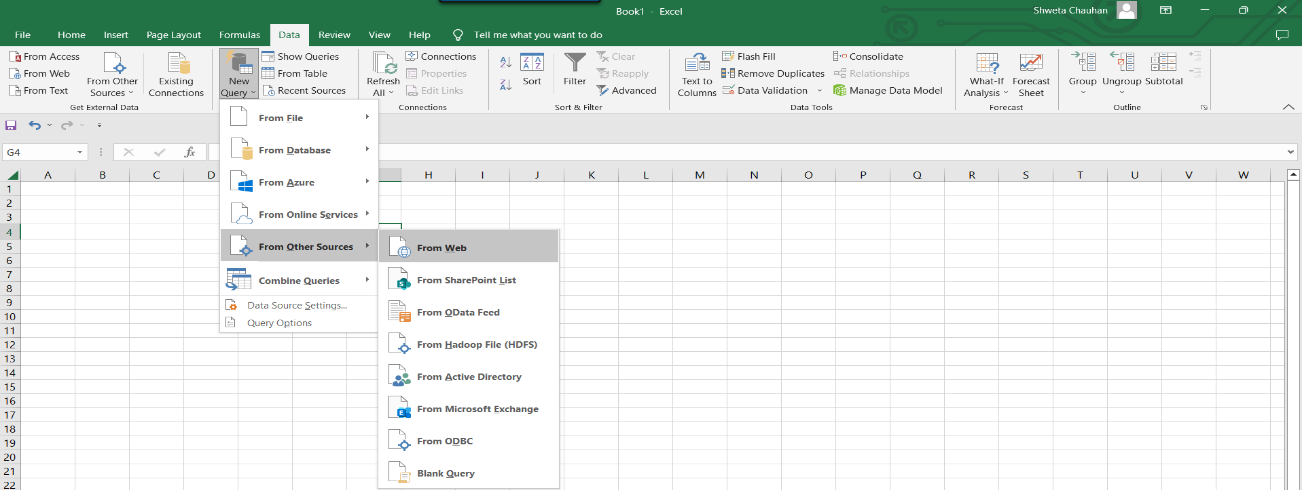
Graphical user interface, text, application

Description automatically generated

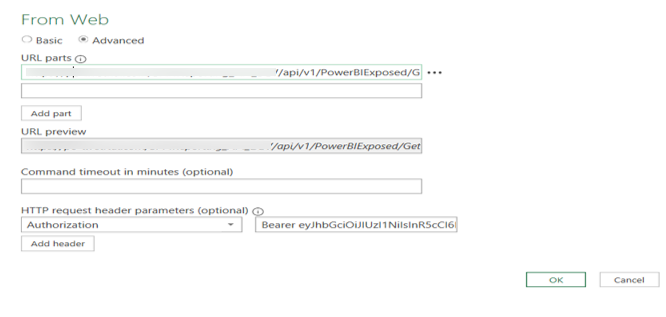
# How to connect to excel

Please follow the below Steps to connect Excel file.

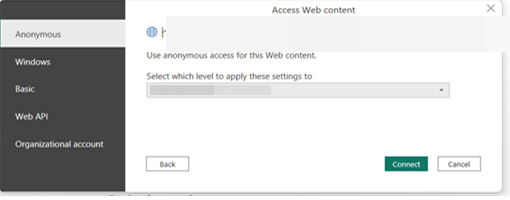
1. Open Excel file, go to the **data** tab.
2. Select “ **from web** ” option from **Get & transform** section/module( Refer to the below screenshot).



1. Choose advance option radio button.
2. Put the **endpoints link** in the Url parts form field.
3. Add header, Fill **authorization** inside HttpRequest header form field, and the field next to the authoriztion with **Bearer** keyword after giving a space paste the copied **token.**
4. Once the details get filled up click **Ok** button(Refer to the below screenshot).



1. After that you will get a window(Refer to the below screenshot), click on **connect** button.



1. Once you will get connected the below shown window will open.

Graphical user interface, application

Description automatically generated

1. Go to the convert tab and select **into table,** the data will shown into a table (refer to the below screenshot).

Graphical user interface, application, Word

Description automatically generated

1. Then select the value which you want to extract, for an example **data column value** from the above table, click on its value(**List**) the vaues will be expand (refer to the below screenshot).

Graphical user interface, application

Description automatically generated

1. Right click over the **list** column select **to table** option.

Graphical user interface, application, table

Description automatically generated

1. Click on the expanded icon on the left side of column, select the columns you want to display and click **ok** button.

Graphical user interface, application, Word

Description automatically generated

1. After expanding the column, you will get your entire details of a record respectively.

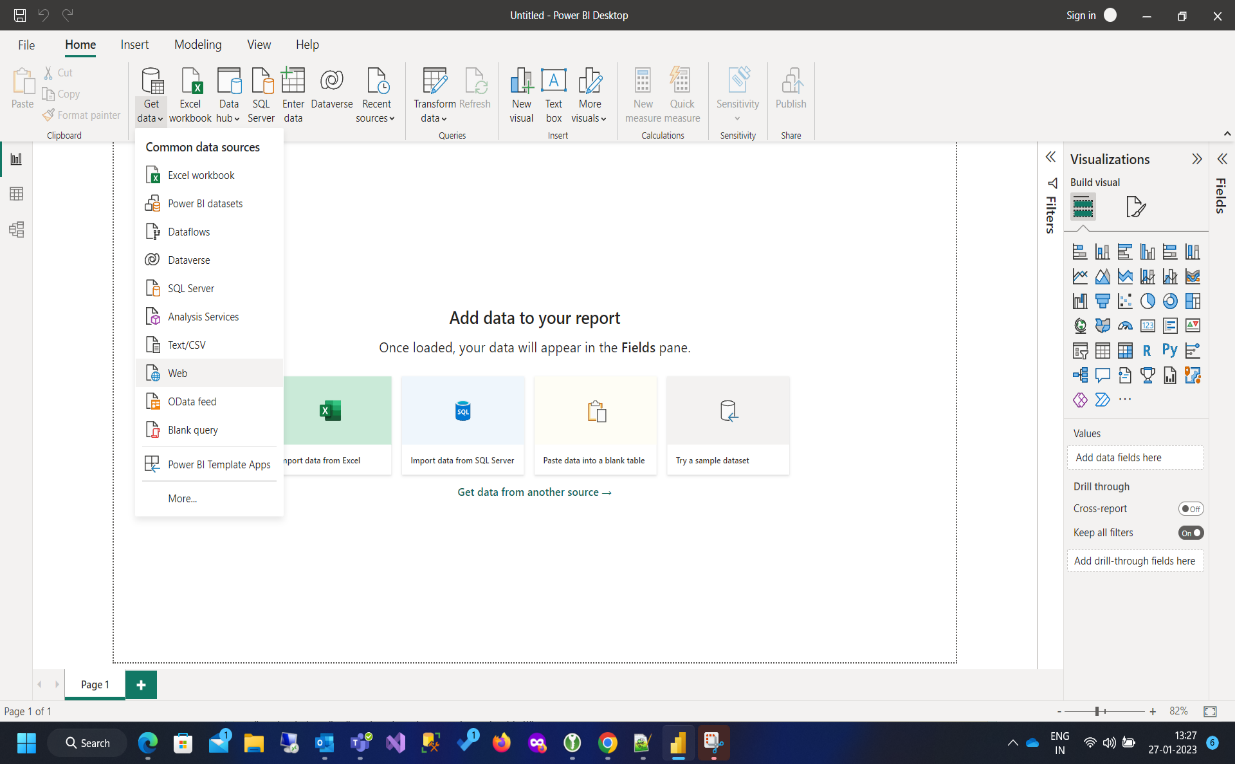
A screenshot of a computer

Description automatically generated with medium confidence

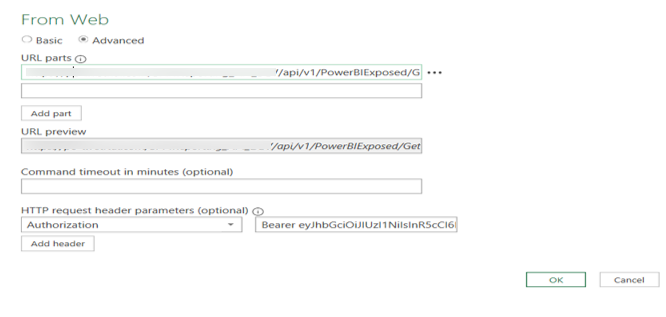
# How to connect to power BI tool

Please follow the below Steps to connect to power BI tool.

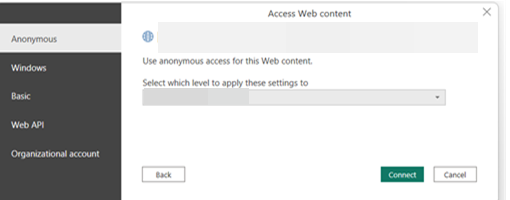
1. Open Excel file, go to the **Get** **data** tab.
2. Select “ **web** ” option from **Data** section/module( Refer to the below screenshot).



1. Choose advance option radio button.
2. Put the **endpoints link** in the Url parts form field.
3. Add header, Fill **authorization** inside HttpRequest header form field, and the field next to the authoriztion with **Bearer** keyword after giving a space paste the copied **token.**
4. Once the details get filled up click **Ok** button(Refer to the below screenshot).



1. After that you will get a window(Refer to the below screenshot), click on **connect** button.



1. Once you will get connected the below shown window will open.

Graphical user interface, application

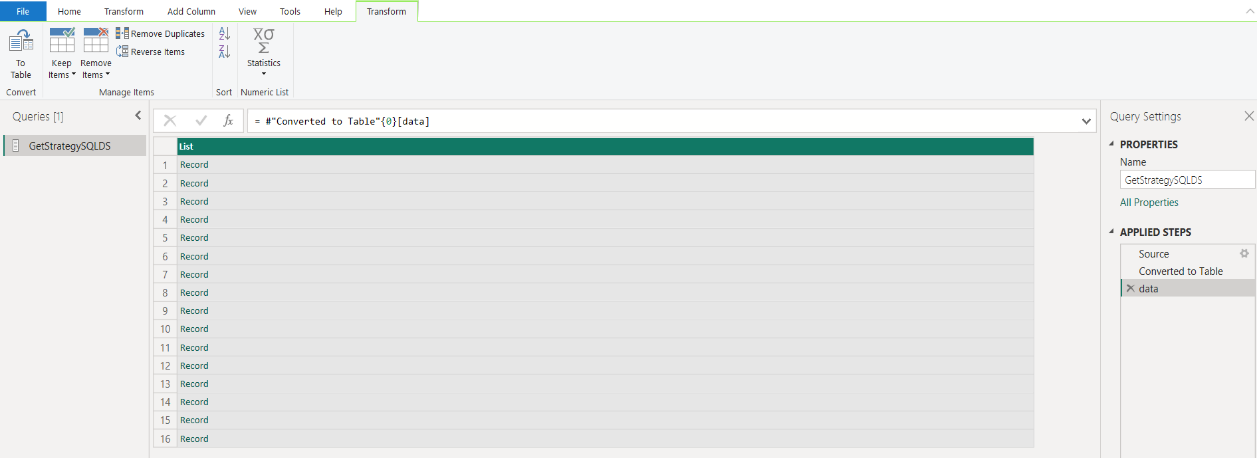
Description automatically generated

1. Go to the convert tab and select **into table,** the data will shown into a table (refer to the below screenshot).

Graphical user interface, text, application

Description automatically generated

1. Then click on the value which you want to extract, for an example **data column value** from the above table, click on its value(**List**) the vaues will be expand (refer to the below screenshot).

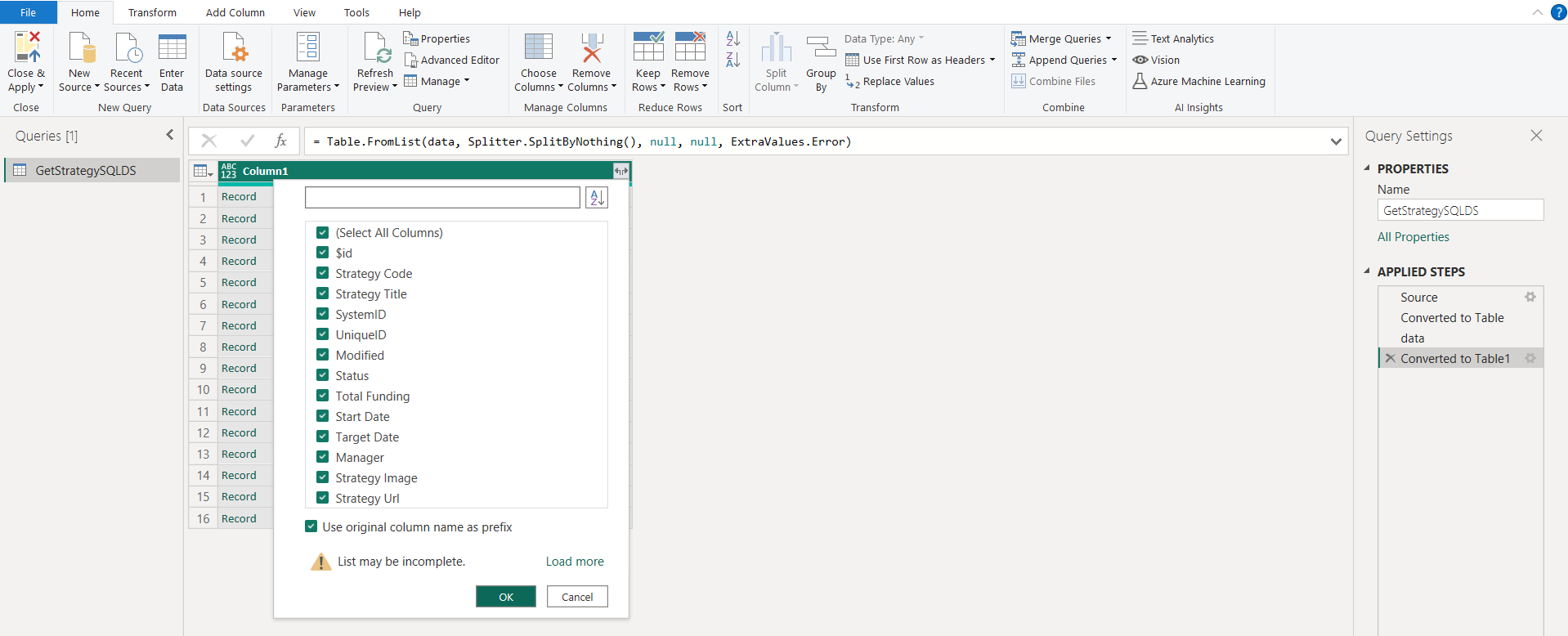


1. Right click over the **list** column select **to table** option.

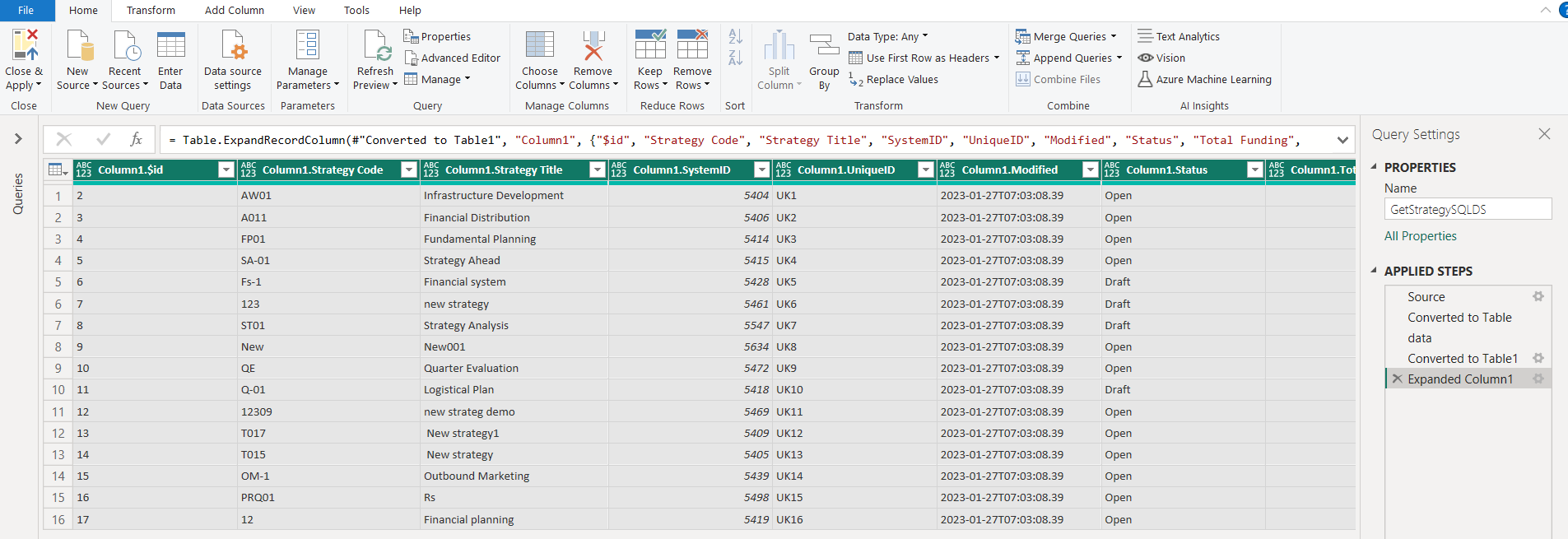
Graphical user interface, application

Description automatically generated with medium confidence

1. Click on the expanded icon on the left side of column, select the columns you want to display and click **ok** button.



1. After expanding the column , you will get your entire details of a record respectively.



# InitiativeRegisterSQLDSView

InitiativeRegister reporting API is a collection of Project Details that is form fields and Register information as shown below in Screen.

Graphical user interface, table

Description automatically generated with medium confidence

**Img – Initiative Register**



**Img – Project “PRQ-000001” Details**

The API includes the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Field Name** | **Field Description** |
| 1 | Benefit Tracking Period | Project Details view field |
| 2 | Benefits | Project Health Indicator for Benefits |
| 3 | Code | Project/Epic code from Details view |
| 4 | Cost Running | Project Running Cost |
| 5 | Creation Date | Project Creation Date |
| 6 | Currency | Project Currency |
| 7 | Effort | Total Project Anticipated Efforts |
| 8 | Finish Date | Project Finish Date |
| 9 | Initiative Manager | Project Manager from Project detail’s view |
| 10 | Initiative Sponser | Project Sponsor from Project detail’s view |
| 11 | Initiative Type | Project Type if Demand or Execution |
| 12 | Is Benefit Tracking | Project Details view field |
| 13 | Originating Department | Project Details view field |
| 14 | Owner | Project Details view field |
| 15 | Process Name | Epic / Project |
| 16 | Product | Product |
| 17 | Project ID | Unique ID |
| 18 | Rating | Project rating |
| 19 | Requested By | Project Requested by |
| 20 | Start Date | Project Start Date |
| 21 | State | Project State |
| 22 | Strategic Priority | Project Details view field |
| 23 | Task Effort | Project Details view field |
| 24 | Title | Project Title |
| 25 | Total anticipated | Project Details view field |
| 26 | Workflow Steps | Project Workflow Step name |
| 27 | WSJF | WSJF rating on Project |

To get Initiative Register View from SPM Reporting API, GetInitiativeRegisterSQLDSView API is called. It is a get request.

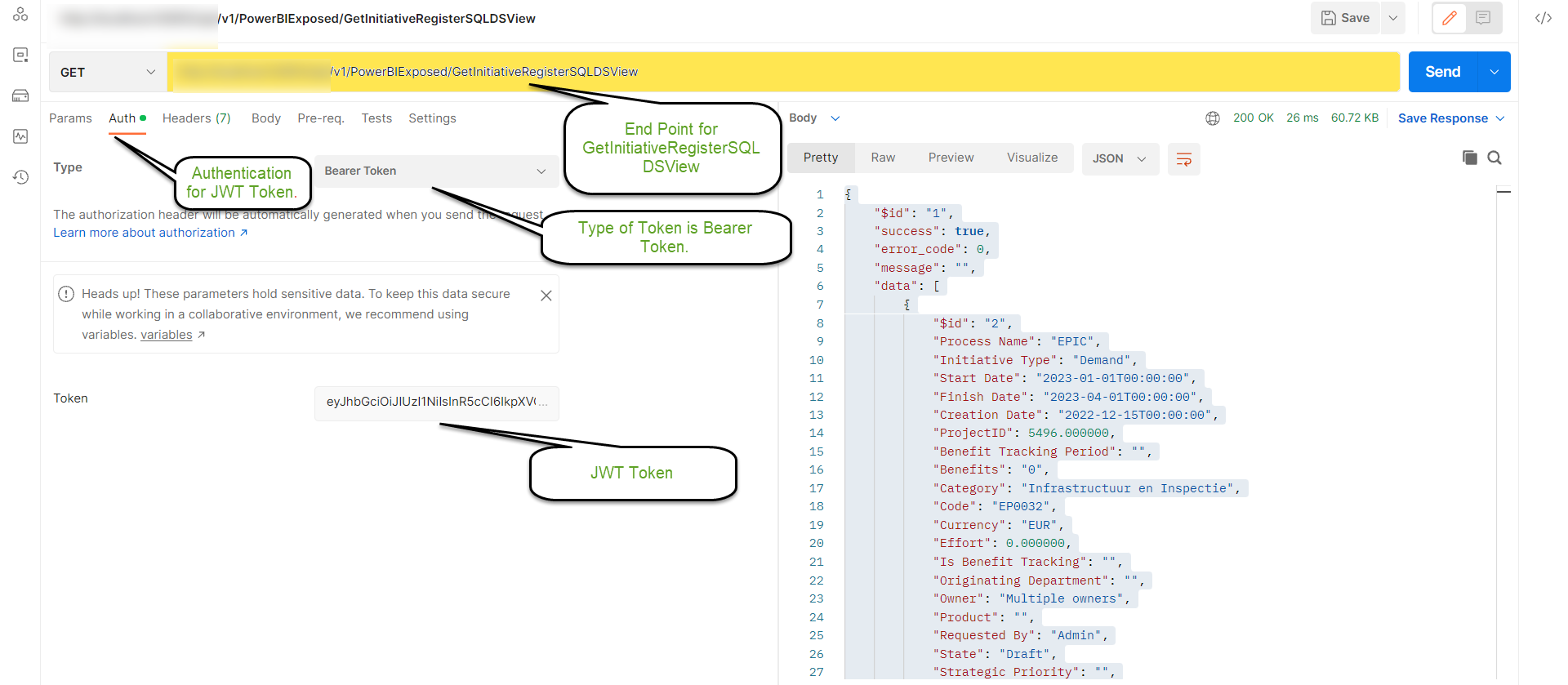
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/**v1/PowerBIExposed/GetInitiativeRegisterSQLDSView** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"Process Name": "EPIC",

"Initiative Type": "Demand",

"Start Date": "2023-01-01T00:00:00",

"Finish Date": "2023-04-01T00:00:00",

"Creation Date": "2022-12-15T00:00:00",

"ProjectID": 5496.000000,

"Benefit Tracking Period": "",

"Benefits": "0",

"Category": "Infrastructuur en Inspectie",

"Code": "EP0032",

"Currency": "EUR",

"Effort": 0.000000,

"Is Benefit Tracking": "",

"Originating Department": "",

"Owner": "Multiple owners",

"Product": "",

"Requested By": "Admin",

"State": "Draft",

"Strategic Priority": "",

"Task Effort": 0.000000,

"Title": "Toevoegen AMEX",

"Initiative Manager": "",

"Initiative Sponser": "",

"Total anticipated cost": 0.000000,

"Rating": "0",

"WSJF": 0.000000,

"Workflow Steps": "Funnel ",

"Last Status Update": **null**

}]}

# PROJECTRegisterEPICSQLDS

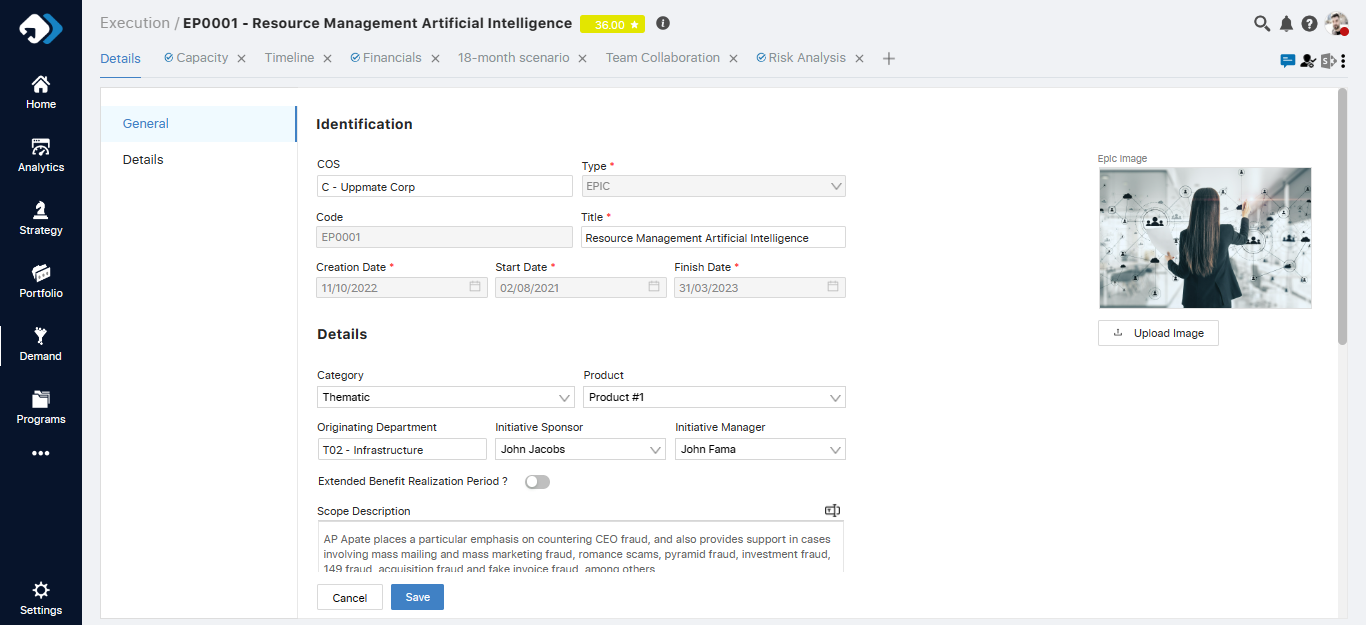
Project Register EPIC reporting API is a collection of Epic Register information & Epic Details form fields and as shown below in Screen.

The fields are available in API based on the existing Project Type Form fields available in Form Builder. As per form updates the API will return values accordingly.

Graphical user interface, application

Description automatically generated

**Img – Epic Register**



**Img – Epic “EP0001” Details**

The API includes the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Field Name** | **Field Description** |
| 1 | Benefit Tracking Period | Epic Details view field |
| 2 | Benefits | Epic Health Indicator for Benefits |
| 3 | Code | Epic code from Details view |
| 4 | Cost of Delay | Epic Running Cost |
| 5 | Creation Date | Epic Creation Date |
| 6 | Currency | Epic Currency |
| 7 | Effort | Total Epic Anticipated Efforts |
| 8 | Finish Date | Epic Finish Date |
| 9 | Initiative Manager | Epic Manager from detail’s view |
| 10 | Initiative Sponser | Epic Sponsor from detail’s view |
| 11 | Demand  Type | Epic Type if Demand or Execution |
| 12 | Is Benefit Tracking | Epic Details view field |
| 13 | Originating Department | Epic Details view field |
| 14 | Owner | Epic Details view field |
| 15 | Process Name | Epic / Project |
| 16 | Product | Product |
| 17 | Project ID | Unique ID |
| 18 | Rating | Epic rating |
| 19 | Requested By | Epic Requested by |
| 20 | Start Date | Epic Start Date |
| 21 | State | Epic State |
| 22 | Strategic Priority | Epic Details view field |
| 23 | Task Effort | Epic Details view field |
| 24 | Title | Epic Title |
| 25 | Total anticipated | Epic Details view field |
| 26 | Workflow Steps | Epic Workflow Step name |
| 27 | WSJF | WSJF rating on Epic |
| 28 | Demand objLink | Redirection Path to Epic Details |
| 29 | Image Link | Reditection Path to Epic Logo |
| 30 | Category | Epic Details view field |
| 31 | obj\_id | Unique ID |
| 32 | Last Status Update |  |
| 33 | Approval Decision |  |

To get ProjectRegisterEpicSQLDS details from SPM Reporting API, GetProjectRegisterEPICSQLDS API is called. It is a get request

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/ **v1/PowerBIExposed/GetProjectRegisterEPICSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |

Graphical user interface, diagram, application

Description automatically generated

## Output

In the case of successful authentication, the below JSON model is returned

{

  "$id": "1",

    "success": **true**,

    "error\_code": 0,

    "message": "",

    "data":[

        {

"$id": "2",

"Process Name": "EPIC",

"Demand Type": "Execution",

"Start Date": "2021-08-01T00:00:00",

"Finish Date": "2022-06-16T00:00:00",

"Creation Date": "2021-07-11T00:00:00",

"Demand objLink": "https://xyz.org ",

"Image Link": " https://www.xyz.org /imgres?imgurl ",

"WSJF": 4.750000,

"Cost of Delay": 38.000000,

"State": "In Progress",

"ProjectID": 5122.000000,

"Benefit Tracking Period": "",

"Benefits": 0.000000,

"Category": "",

"Code": "EP0002",

"Currency": "EUR",

"Effort": 0.000000,

"Initiative Manager": "Jamie Sason",

"Initiative Sponser": "John Jacobs",

"Is Benefit Tracking": "false",

"obj\_id": "91",

"Originating Department": "AI Team",

"Owner": "Not Specified",

"Product": "",

"Rating": 0.000000,

"Requested By": "John Fama",

"Strategic Priority": "",

"Task Effort": 0.000000,

"Title": "Document Integration HUB",

"Total anticipated cost": 0.000000,

"Workflow Steps": "Done",

"Approval Decision": "",

"Last Status Update": "2023-01-22T10:04:16.13",

"Sponsor": **null**

        }

]

}

# PROJECTRegisterPROJECTSQLDS

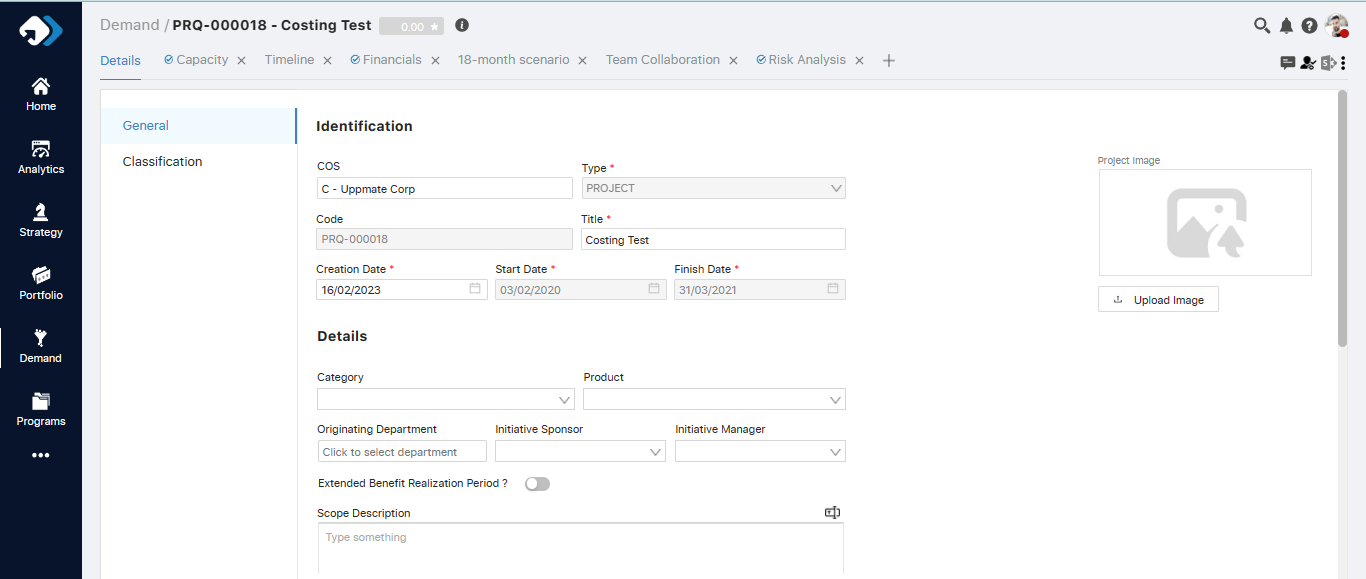
Project Register reporting API is a collection of Epic Register information & Project Details form fields and as shown below in Screen.

The fields are available in API based on the existing Project Type Form fields available in Form Builder. As per form updates the API will return values accordingly.

Graphical user interface, application, email

Description automatically generated

**Img – Project Register**



**Img – “PRQ-000018” Details**

The API includes the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Field Name** | **Field Description** |
| 1 | Process Name | Epic / Project |
| 2 | Demand Type | Project Type if Demand or Execution |
| 3 | Start Date | Project Start Date |
| 4 | Finish Date | Project Finish Date |
| 5 | Creation Date | Project Creation Date |
| 6 | Demand objLink |  |
| 7 | WSJF | WSJF rating on Project |
| 8 | Cost of Delay | Project Running Cost |
| 9 | State | Project State |
| 10 | ProjectID | Unique ID |
| 11 | Benefit Tracking Period | Project Details view field |
| 12 | Benefits | Project Health Indicator for Benefits |
| 13 | Category | Project Details view field |
| 14 | Code | Project/Epic code from Details view |
| 15 | Currency | Project Currency |
| 16 | Effort | Total Project Anticipated Efforts |
| 17 | Initiative Manager | Project Manager from Project detail’s view |
| 18 | Initiative Sponser | Project Sponsor from Project detail’s view |
| 19 | Is Benefit Tracking | Project Details view field |
| 20 | obj\_id | Project Start Date |
| 21 | Originating Department | Project Details view field |
| 22 | Owner | Project Details view field |
| 23 | Product | Product |
| 24 | Rating | Project rating |
| 25 | Requested By | Project Requested by |
| 26 | Strategic Priority | Project Details view field |
| 27 | Task Effort | Project Details view field |
| 28 | Title | Project Title |
| 29 | Workflow Steps | Project Workflow Step name |

To get ProjectRegisterProjectSQLDS details from SPM Reporting API, GetProjectRegisterPROJECTSQLDS API is called. It is a get request

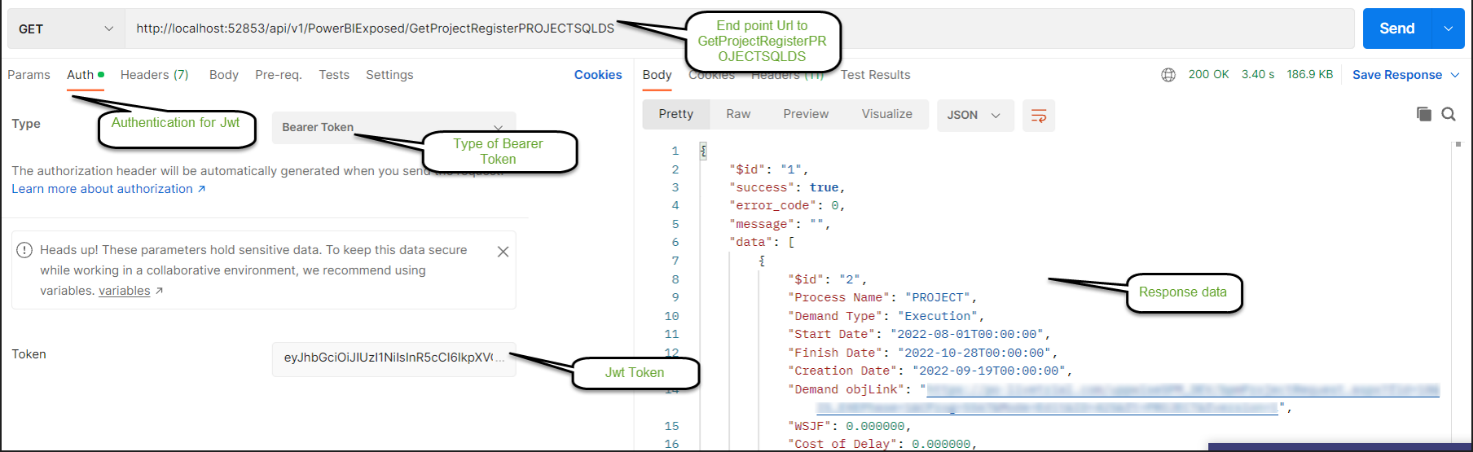
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/ **v1/PowerBIExposed/** **GetProjectRegisterPROJECTSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

    "success": **true**,

    "error\_code": 0,

    "message": "",

    "data": [

        {

            "$id": "2",

            "Process Name": "PROJECT",

            "Demand Type": "Execution",

            "Start Date": "2022-08-01T00:00:00",

            "Finish Date": "2022-10-28T00:00:00",

            "Creation Date": "2022-09-19T00:00:00",

            "Demand objLink": "https://xyz.org",

            "WSJF": 0.000000,

            "Cost of Delay": 0.000000,

            "State": "Draft",

            "ProjectID": 5567.000000,

            "Benefit Tracking Period": "",

            "Benefits": 0.000000,

            "Category": "",

            "Code": "PRQ-000079",

            "Currency": "EUR",

            "Effort": 516.000000,

            "Initiative Manager": "",

            "Initiative Sponser": "",

            "Is Benefit Tracking": "",

            "obj\_id": "425",

            "Originating Department": "",

            "Owner": "Global Administrator",

            "Product": "",

            "Rating": 0.000000,

            "Requested By": "Admin",

            "Strategic Priority": "",

            "Task Effort": 0.000000,

            "Title": "Test Timesheet",

            "Total anticipated cost": 8300.000000,

            "Workflow Steps": "Intake Info Revision",

            "CAPEX Budget ConsCurr": **null**,

            "CAPEX Budget LC": **null**,

            "CAPEX Budget OptCurr": **null**,

            "Cost of Delay1": **null**,

            "Cost Running": **null**,

            "State1": **null**,

            "Business Process Owner": **null**,

            "Process Current Opex(EUR/Year)": **null**,

            "Responsible Department": **null**,

            "Classification": **null**,

            "Maturity Level": **null**,

            "Business Process Assessment": **null**,

            "Parent Business Capability": **null**,

            "Application": **null**,

            "Last Seen": **null**,

            "Last Status Update": **null**,

            "Priority": **null**,

            "Business Capability Owner": **null**,

            "Capability Current Opex(EUR/Year)": **null**,

            "Business Domain": **null**,

            "Capability Assessment": **null**,

            "BusinessProcess": **null**,

       }

        },

# InitiativeMapRegisterSQLDs

InitiativeMap reporting is a connection API contains every module Connection ID’s from Project to Strategy to Portfolios to features and integrated Project management tool (JIRA / CWM) as available in rest of the API’s

The API includes the below listed fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** |  | **Field Name** | **Field Description** |
| 1 |  | Connection Id | Connection ID |
| 2 |  | Current Scenario | Current Scenario |
| 3 |  | CWM / Jira | Project Management Tool |
| 4 |  | Initiative Code | Project / Epic code |
| 5 |  | KeyGoal Code | Key Goal Code |
| 6 |  | Modified | Date Modified |
| 7 |  | Objective Code | Objective Code in Objective widget |
| 8 |  | Porftolio Scenario Title | Portfolio Scenario Title |
| 9 |  | Portfolio Code | Portfolio code |
| 10 |  | ProjectId | Unique ID |
| 11 |  | Strategy Code | Code of Strategy |
| 13 |  | Theme Code | Theme Code in theme widget |
| 14 |  | UniqueID | Unique ID |
| 15 |  | Value Stream Code | Value Stream Code |

To get Initiative Map Registers detail from SPM Reporting API, GetInitiativeMapRegisterSQLDS API is called. It is a get request.

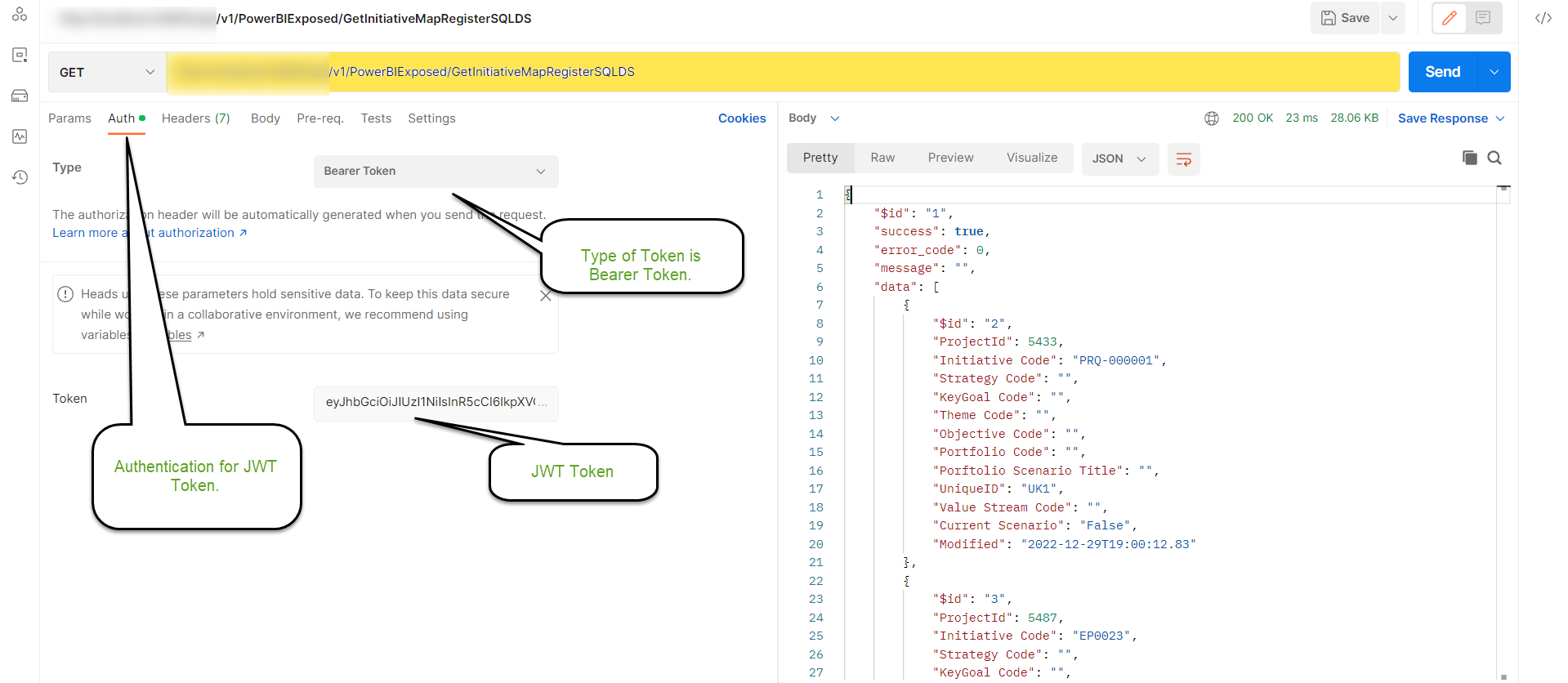
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/**v1/PowerBIExposed/GetInitiativeMapRegisterSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

Token- for the authentication to get the exposed data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"ProjectId": 5433,

"Initiative Code": "PRQ-000001",

"Strategy Code": "",

"KeyGoal Code": "",

"Theme Code": "",

"Objective Code": "",

"Portfolio Code": "",

"Porftolio Scenario Title": "",

"UniqueID": "UK1",

"Value Stream Code": "",

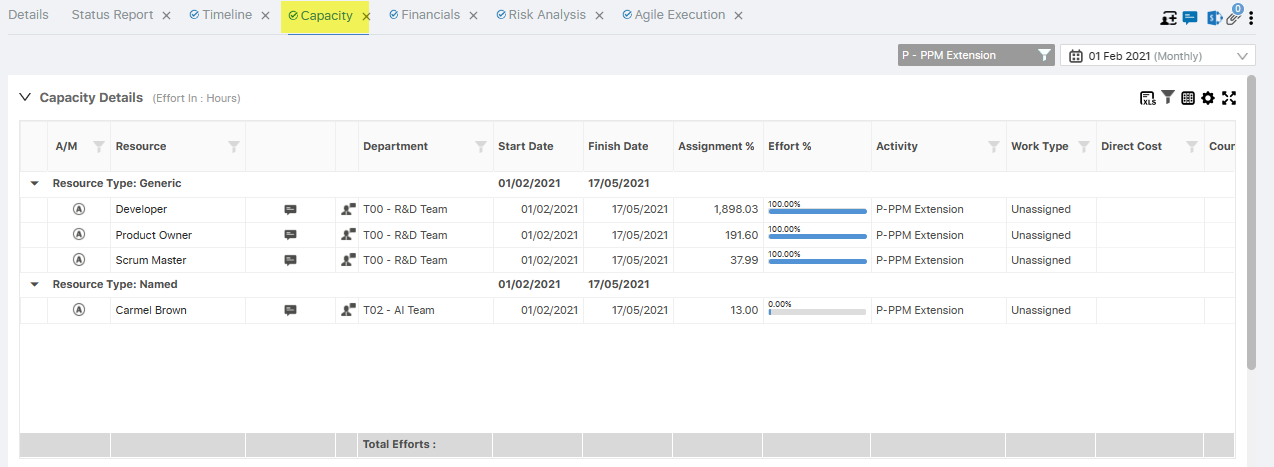
"Current Scenario": "False",

"Modified": "2022-12-29T19:00:12.83"

}]}

# ResourceAvailabilitySQLDS

ResourceAvailability reporting API contains the information related to the Resources assigned over the project, majorly the Resource effort and Availability. The Source of API is the Capacity view from project workspace as shown below.



**Img – Project/Epic Workspace > Capacity View**

The API contains the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field Name** | **Field Description** |
| 1 | Actual Work In Days | Actual Efforts of Resources in Days invested in Project |
| 2 | Actual Work In Hours | Actual Efforts of Resources in Hours invested in Project |
| 3 | Actual Work In Months | Actual Efforts of Resources in Months invested in Project |
| 4 | Availability % | Total Availability of Resource as per its Resource calendar |
| 5 | Baseline Work By Days | Baseline Efforts of Resources in Days invested in Project |
| 6 | Baseline Work By Hours | Baseline Efforts of Resources in Hours invested in Project |
| 7 | Baseline Work By Months | Baseline Efforts of Resources in Months invested in Project |
| 8 | Department | Name of department where resource belong |
| 9 | Forecast Work By Days | Forecast Efforts of Resources in Days invested in Project |
| 10 | Forecast Work By Hours | Forecast Efforts of Resources in Hours invested in Project |
| 11 | Forecast Work By Months | Forecast Efforts of Resources in Months invested in Project |
| 13 | Initiative Code | Project / Epic code |
| 14 | Initiative Title | Title of Project / Epic |
| 15 | Internal / External | Type of Resources |
| 16 | Location | Geographical Location Of Resource |
| 17 | Month | Data as in monthly Phasing |
| 18 | Named / Generic | Type of Resources |
| 19 | Project Job Title | Title of Project / Epic |
| 20 | ProjectId | Unique ID |
| 21 | Resource Availability Days Per Month | Availability of Resource Days per Month in Project |
| 22 | Resource Availability Hours Per Month | Availability of Resource Hours per Month in Project |
| 23 | Resource Availability Month Per Month | Availability of Resource in Month over the Project |
| 24 | Resource Name | Name Of Resource |
| 25 | Resource Type | Resource Type |
| 26 | Work Type | Type Of work (Unassigned/Normal Time/Research) |
| 27 | Working Hrs / Day | Working Hours per Day of Resource as per Resource Calendar |
| 28 | Year | Annual Phasing |

To get Resource Availability from SPM Reporting API, GetResourceAvailabilitySQLDS API is called.It is called a get request

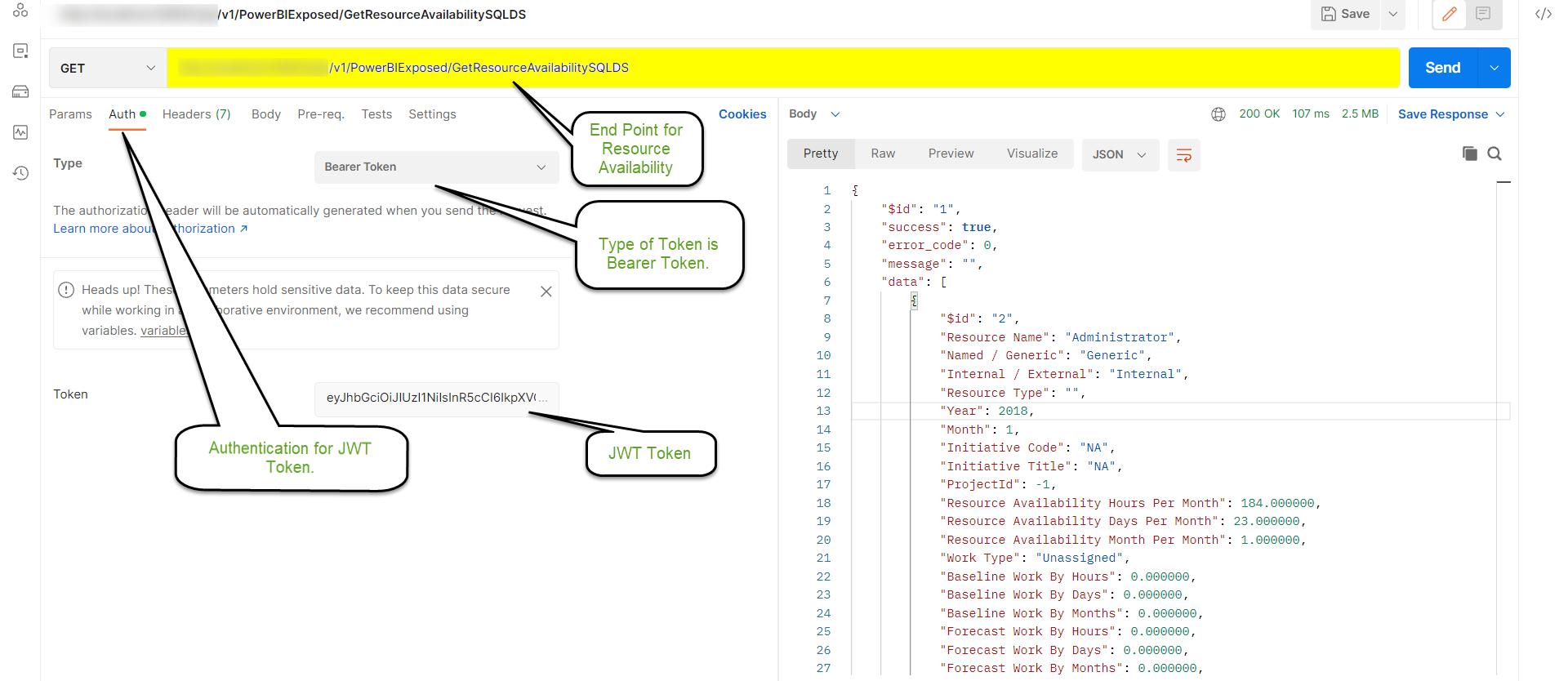
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl **>**/**v1/PowerBIExposed/GetResourceAvailabilitySQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"Resource Name": "Administrator",

"Named / Generic": "Generic",

"Internal / External": "Internal",

"Resource Type": "",

"Year": 2018,

"Month": 1,

"Initiative Code": "NA",

"Initiative Title": "NA",

"ProjectId": -1,

"Resource Availability Hours Per Month": 184.000000,

"Resource Availability Days Per Month": 23.000000,

"Resource Availability Month Per Month": 1.000000,

"Work Type": "Unassigned",

"Baseline Work By Hours": 0.000000,

"Baseline Work By Days": 0.000000,

"Baseline Work By Months": 0.000000,

"Forecast Work By Hours": 0.000000,

"Forecast Work By Days": 0.000000,

"Forecast Work By Months": 0.000000,

"Actual Work In Hours": 0.000000,

"Actual Work In Days": 0.000000,

"Actual Work In Months": 0.000000,

"Department": "",

"Project Job Title": "",

"Working Hrs / Day": 8,

"Availability %": 100,

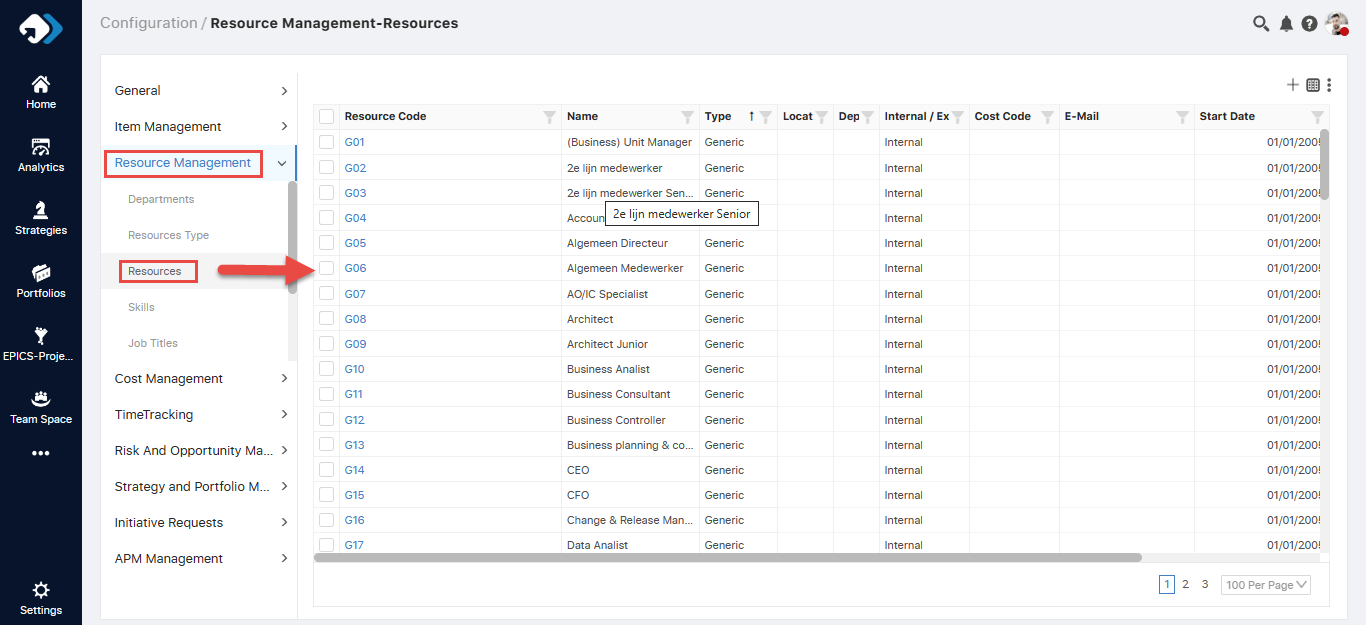
"Location": ""

}]}

# ResourceMapSQLDS

The ResourceMap reporting API contains the Resource Details from the Standard codes

of Application Settings.



**Img –Resource Details**

The API contains the fields listed below.

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Field Name** | **Field Description** |
| 1 | Department | Name of Department |
| 2 | Department Manager | Department Manager Name |
| 3 | Field1 | Custom field (if any) |
| 4 | Field2 | Custom field (if any) |
| 5 | Field3 | Custom field (if any) |
| 6 | Field4 | Custom field (if any) |
| 7 | Field5 | Custom field (if any) |
| 8 | Initiative Code | Project / Epic code |
| 9 | Internal / External | Type of Resources |
| 10 | Named / Generic | Type of Resources |
| 11 | ProjectId | Unique ID |
| 12 | Resource Type | Resource Type |
| 13 | ResourceCode | ResourceCode |
| 14 | ResourceName | Name Of Resource |
| 15 | Skill | Skill of Resource |

To get Feature SQLDS details from SPM Reporting API, GetResourceMapSQLDS API is called. It is a get request.

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/ **v1/PowerBIExposed/GetResourceMapSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |

Graphical user interface, text, application

Description automatically generated

## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"ProjectId": 5121,

"Initiative Code": "EP0001",

"ResourceCode": "G01",

"ResourceName": "Product Owner",

"Department": "R&D Team",

"Department Manager": "Carter May",

"Named / Generic": "Generic",

"Internal / External": "Internal",

"Resource Type": "",

"Field1": "",

"Field2": "",

"Field3": "",

"Field4": "",

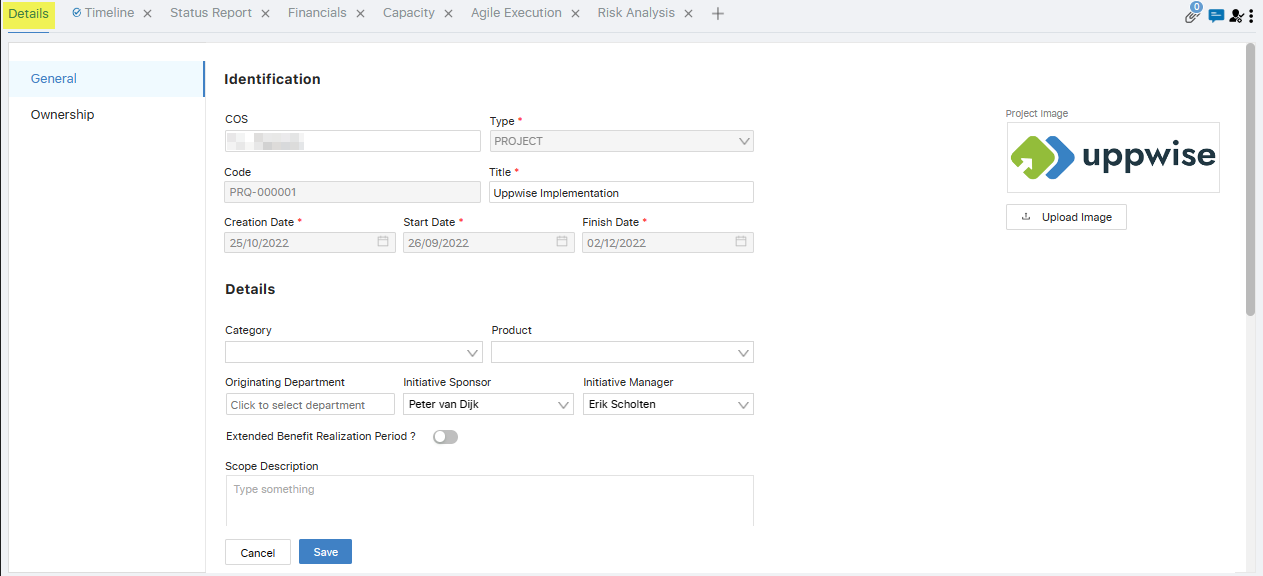
"Field5": "",

"Skill": ""

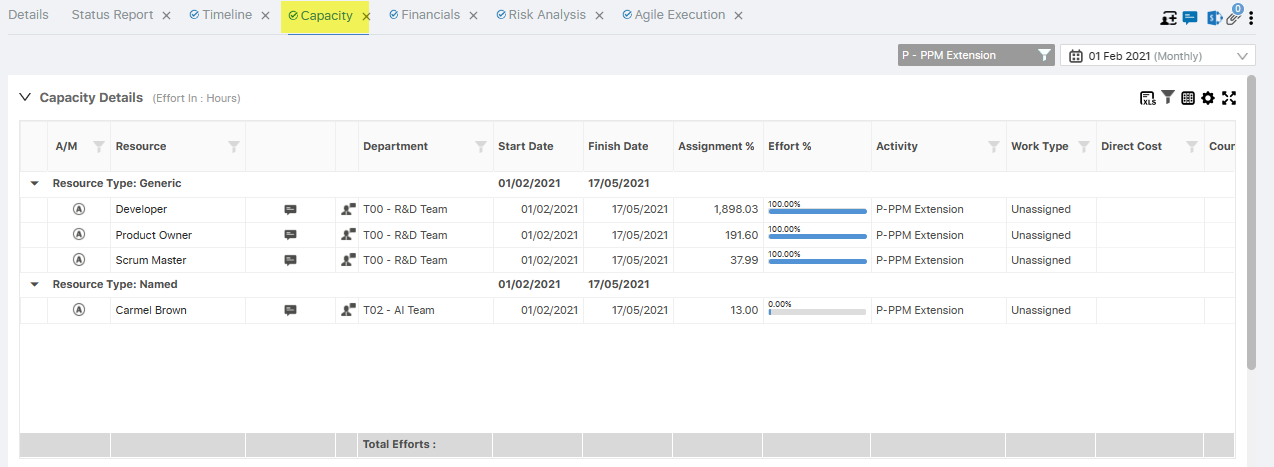
},

# InitiativeEffortDetailsbyPeriod

InitiativeEffortDetailsByPeriod reporting API is a combination of both Project details along with the Resources working over the projects Annually.



**Img – Project Details**



**Img – Capacity view**

The API includes the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field Name** | **Field Description** |
| 1 | Actual Work In Days | Actual Work of Resources in Days invested in Project |
| 2 | Actual Work In Hours | Actual Work of Resources in Hours invested in Project |
| 3 | Actual Work IN Months | Actual Work of Resources in Months invested in Project |
| 4 | Availability % | Total Availability of Resource as per its Resource calendar |
| 5 | Baseline Work By Days | Baseline work of Resources in Days invested in Project |
| 6 | Baseline Work By Hours | Baseline Work of Resources in Hours invested in Project |
| 7 | Baseline Work By Months | Baseline Work of Resources in Months invested in Project |
| 8 | Benefit Tracking Period | Project Details view field |
| 9 | Benefits | Project Health Indicator for Benefits |
| 10 | Category | Project Details view field |
| 11 | Creation Date | Project Creation Date |
| 12 | Currency | Project Currency |
| 13 | Department | Department |
| 14 | Effort | Effort |
| 15 | Finish Date | Finish Date |
| 16 | Forecast Work By Days | Forecast Work of Resources in Days invested in Project |
| 17 | Forecast Work By Hours | Forecast Work of Resources in hours invested in Project |
| 18 | Forecast Work By Months | Forecast Work of Resources in Months invested in Project |
| 19 | Initiative Manager | Initiative Manager Name |
| 20 | Initiative Sponser | Initiative Sponser Name |
| 21 | Initiative Type | Initiative Type |
| 22 | Internal / External | Type of Resources |
| 23 | Is Benefit Tracking | Is Benefit Tracking |
| 24 | Month | Month |
| 25 | Named / Generic | Type of Resources |
| 26 | Originating Department | Originating Department |
| 27 | Owner | Owner |
| 28 | Period | Period |
| 29 | Portfolio Code | Portfolio code |
| 30 | Portfolio Title | Portfolio type |
| 31 | ProjectID | Project ID |
| 32 | Rating | Rating |
| 33 | Requested By | Requested By |
| 34 | Resource Availability Days Per Month | Availability of Resource Days per Month in Project |
| 35 | Resource Availability Hours Per Month | Availability of Resource Hours per Month in Project |
| 36 | Resource Availability Month Per Month | Availability of Resource in Month over the Project |
| 37 | Resource Name | Name Of Resource |
| 38 | Resource Type | Resource Type |
| 39 | Start Date | Project Start Date |
| 40 | State | Project State |
| 41 | Strategic Priority | Project Details view field |
| 42 | Task Effort | Project Details view field |
| 43 | Total anticipated cost | Project Details view field |
| 44 | Work Type | Type Of work (Unassigned/Normal Time/Research) |
| 45 | Workflow Steps | Project Workflow Step name |
| 46 | Working Hrs / Day | Working Hours per Day of Resource as per Resource Calendar |
| 47 | WSJF | WSJF rating on Project |
| 48 | Year | Year of Phasing as per Cost & Capacity |

To get Initiative Effort Details by Period from SPM Reporting API, GetvwInitiativeEffortDetailsbyPeriod API is called. It is a get request.

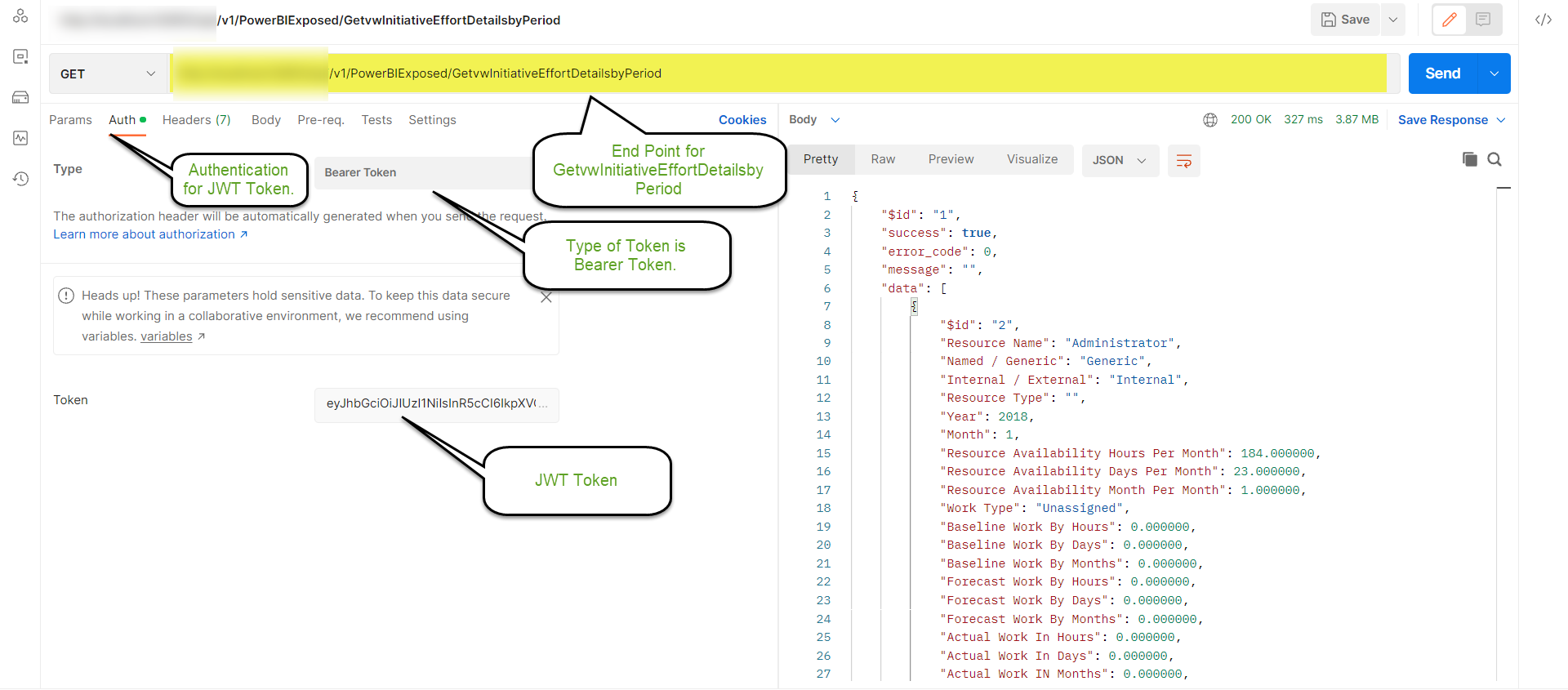
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/**v1/PowerBIExposed/GetvwInitiativeEffortDetailsbyPeriod** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"Resource Name": "Administrator",

"Named / Generic": "Generic",

"Internal / External": "Internal",

"Resource Type": "",

"Year": 2018,

"Month": 1,

"Resource Availability Hours Per Month": 184.000000,

"Resource Availability Days Per Month": 23.000000,

"Resource Availability Month Per Month": 1.000000,

"Work Type": "Unassigned",

"Baseline Work By Hours": 0.000000,

"Baseline Work By Days": 0.000000,

"Baseline Work By Months": 0.000000,

"Forecast Work By Hours": 0.000000,

"Forecast Work By Days": 0.000000,

"Forecast Work By Months": 0.000000,

"Actual Work In Hours": 0.000000,

"Actual Work In Days": 0.000000,

"Actual Work IN Months": 0.000000,

"Department": "",

"Working Hrs / Day": 8,

"Availability %": 100,

"Period": "01/01/2018 ",

"ProjectID": -1.000000,

"Initiative Type": "",

"Start Date": **null**,

"Finish Date": **null**,

"Creation Date": **null**,

"Benefit Tracking Period": "",

"Benefits": "0",

"Category": "",

"Total anticipated cost": 0.000000,

"Currency": "",

"Effort": 0.000000,

"Is Benefit Tracking": "",

"Originating Department": "",

"Owner": "",

"Requested By": "",

"State": "",

"Strategic Priority": "",

"Task Effort": 0.000000,

"Initiative Manager": "",

"Initiative Sponser": "",

"Rating": "0",

"WSJF": 0.000000,

"Workflow Steps": "",

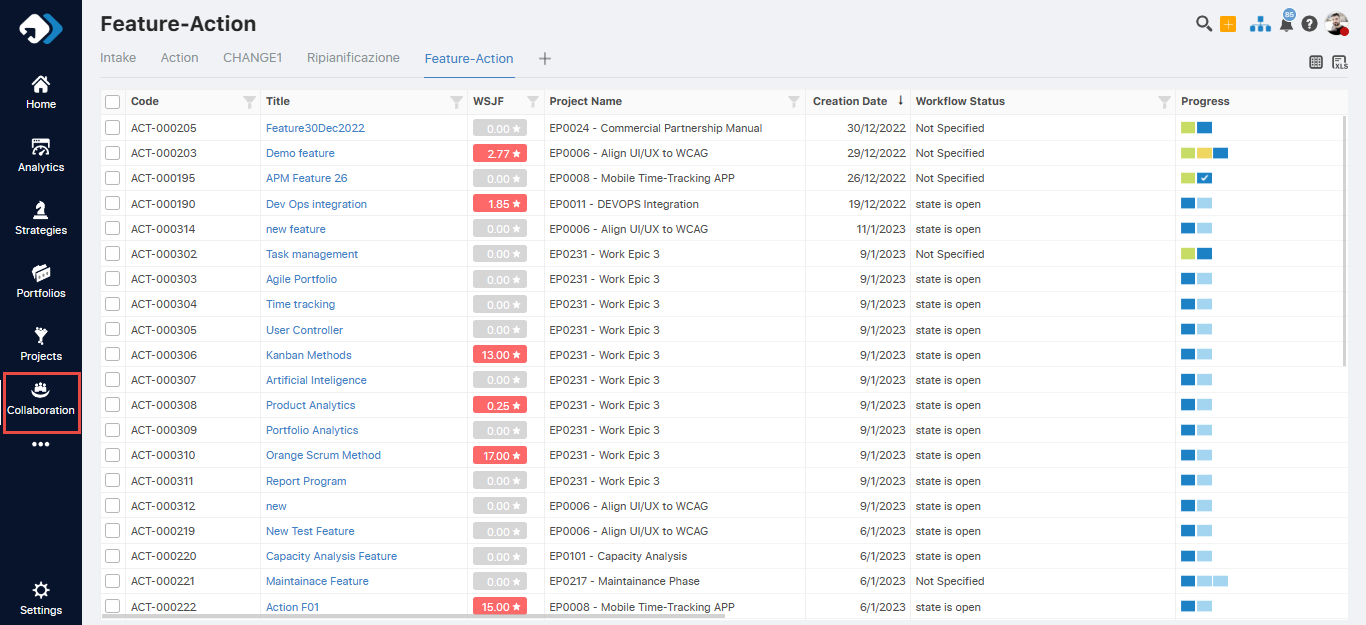
"Portfolio Code": "",

"Portfolio Title": ""

}]}

# FeatureSQLDS

The FeatureSQLDS reporting API Contains the Feature details associated with Projects and created from Collaboration module of application. These features then further connected to any Project Management tools that is JIRA or CWM.



**Img – Collaboration Feature Register**

Graphical user interface

Description automatically generated

**Img –Feature Details**

The API contains the fields listed below.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field Name** | **Field Description** |
| 1 | Anticipated Efforts | Calculated efforts |
| 2 | Connection Id | Connection Id of JIRA or CWM |
| 3 | Feature Code | Feature Code |
| 4 | Feature Title | Title of Feature |
| 5 | Feature WSJF Scoring | Feature WSJF Scoring |
| 6 | Jira / CWM | Jira / CWM |
| 7 | Product | Product |
| 8 | Project Code | Project Code or Epic Code |
| 9 | ProjectId | ProjectId |
| 10 | Reference Activity | Reference Activity |
| 11 | Responsible | Responsible |
| 12 | Target Date | Target Date |
| 13 | Team | Team |
| 14 | Type | Type |
| 15 | UniqueID | Unique ID |

To get Feature SQLDS details from SPM Reporting API, GetFeatureSQLDS API is called. It is a get request.

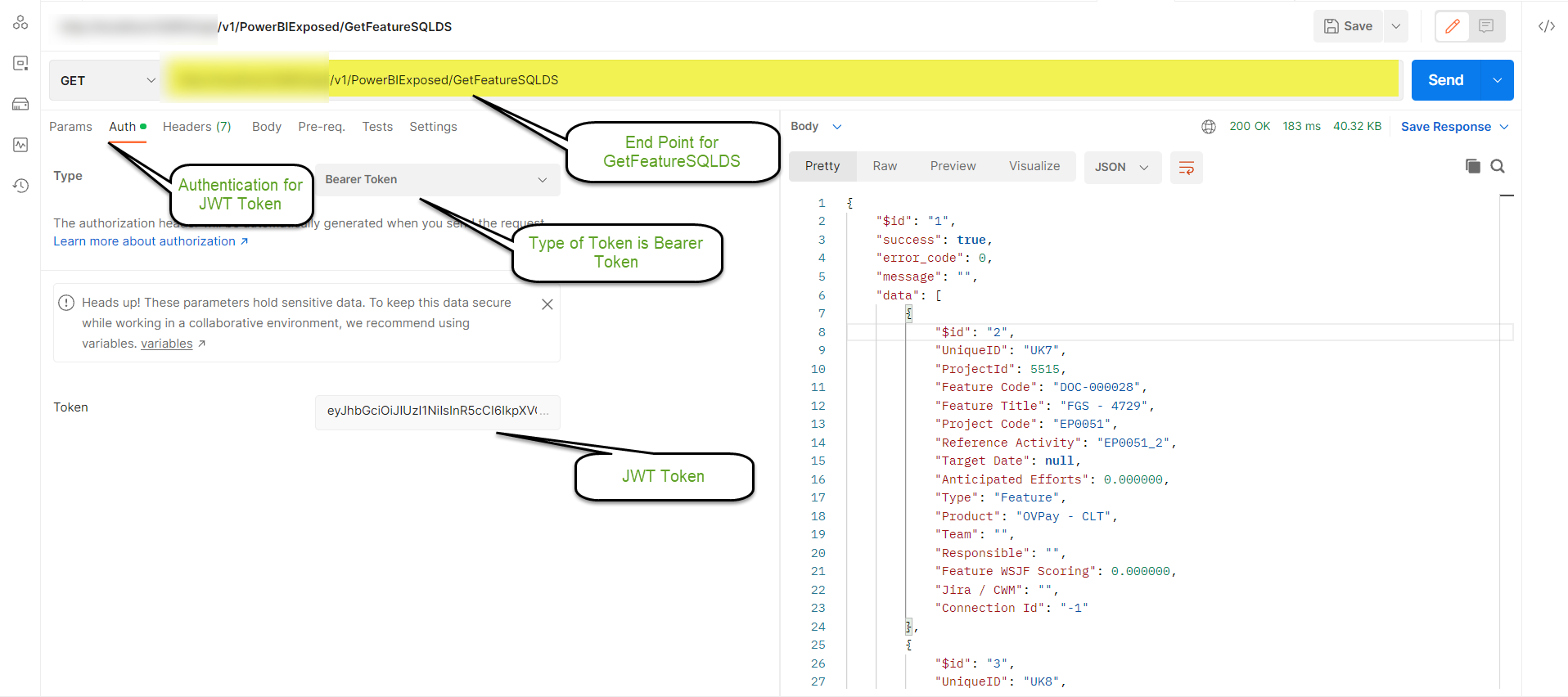
|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**BaseUrl**>**/ **v1/PowerBIExposed/GetFeatureSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"UniqueID": "UK7",

"ProjectId": 5515,

"Feature Code": "DOC-000028",

"Feature Title": "FGS - 4729",

"Project Code": "EP0051",

"Reference Activity": "EP0051\_2",

"Target Date": **null**,

"Anticipated Efforts": 0.000000,

"Type": "Feature",

"Product": "OVPay - CLT",

"Team": "",

"Responsible": "",

"Feature WSJF Scoring": 0.000000,

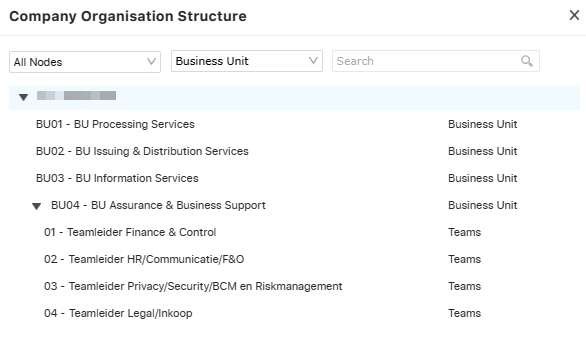
"Jira / CWM": "",

"Connection Id": "-1"

},

# ProjectCOSAssociationSQLDS

ProjectCOSAssociation reporting API contains the Company organization Structure nodes and the associated Project/Epics created over the nodes.



**Img – Company Organization Structure**

The API contains the below listed fields.

|  |  |  |
| --- | --- | --- |
| **S.No** | **Field Name** | **Field Description** |
| 1 | Area/Service | COS Node Level |
| 2 | Company | Company Node |
| 3 | COS Node | Company organization Structure Node where project is created |
| 4 | Facility | Name of COS Node |
| 5 | Group | Name of COS Node |
| 6 | Initiative Code | Project / Epic code |
| 7 | Initiative Process Name | Name of COS Node |
| 8 | Initiative Title | Project / Epic Title |
| 9 | Modified | Modified Date |
| 10 | Project | Name of COS Node |
| 11 | Project Process Name | Project Process Name |
| 12 | Project Id | Unique Id |
| 13 | Subsystem | Name of Cos Node |
| 14 | Unit | Name of Cos Node |

To get Project COS Association from SPM Reporting API, GetProjectCOSAssociationSQLDS API is called. It is a get request.

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<** Base Url **>**/ **v1/PowerBIExposed/GetProjectCOSAssociationSQLDS** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |

Graphical user interface, diagram

Description automatically generated

## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"ProjectId": 5433,

"Project Process Name": "",

"COS Node": "",

"Project": "",

"Area/Service": "",

"Company": "",

"Facility": " ",

"Group": "",

"Subsystem": "",

"Unit": "",

"Modified": "2022-12-29T19:00:13.89",

"Initiative Process Name": "PROJECT",

"Initiative Code": "PRQ-000001",

"Initiative Title": "Uppwise Implementation",

"Business Area": **null**,

"Business Unit": " ",

"Divisies": **null**,

"Domeinen": **null**,

"Teams": " "

}]}

# ScheduleDataSQLDS

ScheduleData reporting API is a collection of the information related to Project / Epic workspace Schedule view / Gantt Editor Timeline as shown below.

Graphical user interface, table

Description automatically generated

**Img – Project/Epic Workspace > Schedule view**

The API contains below listed fields.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field Name** | **Field Description** |
| 1 | (%) Complete | Percentage Completed |
| 2 | Activity | Name of Activity |
| 3 | Actual Duration | Total duration from actual start to finish |
| 4 | Actual Finish | Actual Finish Date |
| 5 | Actual Start | Actual Start Date |
| 6 | Baseline Duration | Total duration from actual start to finish |
| 7 | Baseline Finish | Baseline finish Date |
| 8 | Baseline Start | Baseline start Date |
| 9 | Duration | Total duration from forecast start to finish |
| 10 | Field1 | Custom field on Schedule view (if any available) |
| 11 | Field2 | Custom field on Schedule view (if any available) |
| 12 | Field3 | Custom field on Schedule view (if any available) |
| 13 | Field4 | Custom field on Schedule view (if any available) |
| 14 | Field5 | Custom field on Schedule view (if any available) |
| 15 | Initiative Code | Project / Epic code |
| 16 | Initiative Title | Title of Project / Epic |
| 17 | Manager | Project / Epic Manager |
| 18 | Physical (%) Complete | Actual completed percentage of activity |
| 19 | Planned (%) Complete | Planned completed percentage of activity |
| 20 | ProjectID | Project ID (unique ID) |
| 21 | Remaining Duration | Remaining Duration to complete the activity |
| 22 | Task Finish | Forecast Finish Date of Task/activity |
| 23 | Task Start | Forecast Start Date of Task |
| 24 | WBS | Work Breakdown Structure Code |
| 25 | Weight (%) | Weight (%) |

## Input Parameters

To get Schedule Data from SPM Reporting API, **GetScheduleDataSQLDS** API is called. It is a get request and the API required

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<BaseUrl>/v1/PowerBIExposed/GetScheduleDataSQLDS** |
| **HTTP Headers** | none |

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |



## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"ProjectID": 5490.000000,

"Initiative Code": "EP0026",

"Initiative Title": "Aangesloten verkoopsysteem vervoerder",

"WBS": "P",

"Activity": "EP0026",

"Task Start": "2023-04-07T08:00:00",

"Task Finish": "2023-06-29T17:00:00",

"Manager": "",

"Duration": 53.000000,

"(%) Complete": 0,

"Planned(%) Complete": 0.000000,

"Physical(%)Complete": 0,

"Baseline Start": **null**,

"Baseline Finish": **null**,

"Baseline Duration": 0.000000,

"Actual Start": **null**,

"Actual Finish": **null**,

"Actual Duration": 0.000000,

"Remaining Duration": 58.000000,

"Weight(%)": 100.000000,

"Field1": "",

"Field2": "",

"Field3": "",

"Field4": "",

"Field5": ""

}]}

# InitiativeSummary

InitiativeSummary reporting API is a Summarized collection of Project/Epic details. It includes Financial, Risk, Key indicators, and Capacity data from Project Workspace.

Chart, timeline, treemap chart

Description automatically generated

**Img – Project/Epic Workspace**

The API contains below listed fields.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Field Name** | **Field Description** |
| 1 | Actual | Actual Project Cost |
| 2 | Actual Finish | Actual Project Finish Date |
| 3 | Actual Start | Actual Project Start Date |
| 4 | Actual Work in Days | Actual Efforts of Resources in Days invested in Project |
| 5 | Actual Work in Hours | Actual Efforts of Resources in Hours invested in Project |
| 6 | Actual Work in Months | Actual Efforts of Resources in Months invested in Project |
| 7 | Baseline Finish | Baseline Project Finish Date |
| 8 | Baseline Start | Baseline Project Start Date |
| 9 | Baseline Work by Days | Baseline Efforts of Resources in Days invested in Project |
| 10 | Baseline Work by Hours | Baseline Efforts of Resources in Hours invested in Project |
| 11 | Baseline Work by Months | Baseline Efforts of Resources in Months invested in Project |
| 12 | Benefit Tracking Period | Project Details view field |
| 13 | Benefits | Project Health Indicator for Benefits |
| 14 | Budget | Project Budget Cost |
| 15 | Category | Project Details view field |
| 16 | Code | Project/Epic code from Details view |
| 17 | COS Node | Company organization Structure Node where project is created |
| 18 | Cost Type | CAPEX / OPEX |
| 19 | costs | Project Health Indicator for Cost Elements |
| 20 | Creation Date | Project Creation Date |
| 21 | Curr Impact Score Average | Current Impact Score average of Project Risk |
| 22 | Curr Likelihood Score Average | Current Likelihood Score average of Project Risk |
| 23 | Curr Risk Score | Current Risk Score of Project |
| 24 | Currency | Project Currency |
| 25 | Currency Type | Type of currency (project, Company & Optional) for multi-currency projects |
| 26 | EAC | Project Forecast Cost (Estimate at completion cost) |
| 27 | Effort | Total Project Anticipated Efforts |
| 28 | Element Type | Type of Financial elements is Cost / Benefit |
| 29 | ETC | Project Estimate to Complete cost |
| 30 | Expenditure | Total Project expenditure |
| 31 | Finish Date | Project Finish Date |
| 32 | Forecast Work by Days | Forecast Efforts of Resources in Days invested in Project |
| 33 | Forecast Work by Hours | Forecast Efforts of Resources in Hours invested in Project |
| 34 | Initiative Manager | Project Manager from Project detail’s view |
| 35 | Initiative Sponser | Project Sponsor from Project detail’s view |
| 36 | Initiative Type | Project Type if Demand or Execution |
| 37 | Is Benefit Tracking | Project Details view field |
| 38 | IsBasicCostApplied | Project Details view field |
| 39 | Orig Impact Score | Original Impact Score of Project Risk |
| 40 | Orig Likelihood Score Average | Original Likelihood Score average of Project Risk |
| 41 | Orig Risk Score Average | Original Impact Score average of Project Risk |
| 42 | Originating Department | Project Details view field |
| 43 | Overall | Project Health Indicator |
| 44 | Owner | Project Details view field |
| 45 | Process Name | Epic / Project |
| 46 | ProjectID | Unique ID |
| 47 | Rating | Project rating |
| 48 | Requested By | Project Requested by |
| 49 | Resources | Project Health Indicator |
| 50 | Schedule | Project Health Indicator |
| 51 | Start Date | Project Start Date |
| 52 | State | Project State |
| 53 | Strategic Priority | Project Details view field |
| 54 | Task Effort | Project Details view field |
| 55 | Title | Project Title |
| 56 | Total anticipated cost | Project Details view field |
| 57 | Workflow Steps | Project Workflow Step name |
| 58 | WSJF | WSJF rating on Project |
| 59 | Year | Year of Phasing as per Cost & Capacity |

To get view data of Initiative Summary in SPM Reporting API, GetvwInitiativeSummary API is called It is a get request and the API parameters are required.

|  |  |
| --- | --- |
| **HTTP Method** | GET |
| **Endpoint** | **<**Base Url**>**/**v1/PowerBIExposed/GetvwInitiativeSummary** |
| **HTTP Headers** | none |

## Input Parameters

The method accepts a bearer token in auth type for the authentication of the user

* Token- for the authentication to get the exposed data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Format** | **Mandatory** | **Default value** |
| {token} | For the authentication to get the exposed data | String | Yes | Please consider the Default API Parameters Table |

Graphical user interface, diagram

Description automatically generated

## Output

In the case of successful authentication, the below JSON model is returned

{

"$id": "1",

"success": **true**,

"error\_code": 0,

"message": "",

"data": [

{

"$id": "2",

"Process Name": "PROJECT",

"ProjectID": 5433.000000,

"Initiative Type": "Demand",

"Start Date": "2022-09-26T00:00:00",

"Finish Date": "2022-12-02T00:00:00",

"Creation Date": "2022-10-25T00:00:00",

"Benefit Tracking Period": "",

"Benefits": "0",

"Category": "",

"Code": "PRQ-000001",

"Total anticipated cost": 0.000000,

"Currency": "EUR",

"Effort": 0.000000,

"Is Benefit Tracking": "false",

"Originating Department": "",

"Owner": "Multiple owners",

"Requested By": "Admin",

"State": "In Progress",

"Strategic Priority": "",

"Task Effort": 0.000000,

"Title": "Uppwise Implementation",

"Initiative Manager": "Erik Scholten",

"Initiative Sponser": "Peter van Dijk",

"Rating": "0",

"WSJF": 0.000000,

"Workflow Steps": "Execution",

"COS Node": "",

"Baseline Work By Hours": "2022-10-25T12:36:59.99",

"Baseline Work By Days": **null**,

"Baseline Work By Months": **null**,

"Forecast Work By Hours": **null**,

"Forecast Work By Days": **null**,

"Actual Work In Hours": **null**,

"Actual Work In Days": **null**,

"Actual Work In Months": **null**,

"Currency Type": **null**,

"Year": "ConsCurr",

"Element Type": 2022,

"Cost Type": "Cost",

"IsBasicCostApplied": "Capex",

"Budget": "True",

"EAC": 20168.000000,

"Actual": 20168.000000,

"ETC": 0.000000,

"Expenditure": 20168.000000,

"Overall": 0.000000,

"Schedule": **null**,

"costs": "GREEN",

"Resources": "GREEN",

"Curr Impact Score Average": "YELLOW",

"Curr Likelihood Score Average": **null**,

"Curr Risk Score Average": **null**,

"Orig Impact Score": **null**,

"Orig Likelihood Score Average": **null**,

"Orig Risk Score Average": **null**

}]}