Schedule 8.3.0-8.6.0

30/06/2025 4:36 pm BST

relates to version

Tags: 8.3

Versions

Multiple versions of this API exist:

-
- *8.0.0 - 8.2.x*
- *8.3.0 - 8.6.x: This document*
- *8.7.0+*

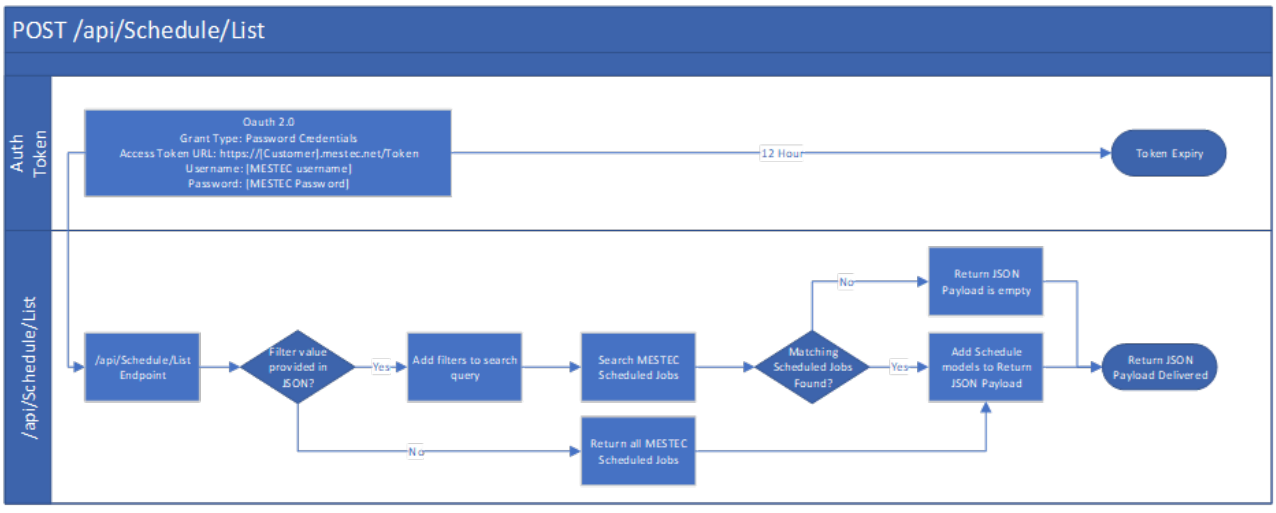
In Eyelit MES-M, 'Schedule' refers to a scheduled job, also referred to as a works order, production order or job.

List

The Schedule – List API call reads existing Schedule Jobs from Eyelit MES-M. The data can be filtered based on the parameters passed in the body of the JSON packet giving the flexibility to search for one or multiple Scheduled Jobs.

Using the Schedule – List API call has no impact on the data within the given Eyelit MES-M application, it is read-only.

Figure 1 - Logic within Schedule/List API Call



Data Prerequisites

There is no data required in Eyelit MES-M to act as a pre-requisite to make the API call to list Scheduled Jobs. If parameters are passed in through the body that return no valid results, an empty payload will be returned.

Request

Table 2 shows the method and endpoint required to make the API call to list Scheduled Jobs.

Table 2 - Outbound Message Detail for Schedule - List

Table 2 - Outbound Message Detail for Schedule - List

Method	URL Structure	Endpoint
POST	https://[environment].mestec.net	/api/Schedule/List

The body of the payload should follow the format below.

```
{
  "id": 0,
  "orderNo": "string"
}
```

See Table 3 for information on which fields are optional, the appropriate data types and the mappings to fields in Eyelit MES-M.

Table 3 - Parameter Information for Schedule - List

Parameter Name	Data Mapping	Data Type	Mandatory	Case Sensitive	Match Type
id	schedule.id	NUMBER	No	N/A	Exact
orderno	schedule.orderNo	VARCHAR2(250 BYTE)	No	No	Exact

Note: For any fields where the match type is 'Like', a percent symbol should be used as a wildcard character to indicate a number of characters within the given string.

Sample Request

See below for sample use cases with examples of the JSON payload format required.

To list all Scheduled Jobs:

```
{

}
```

To list Scheduled Jobs where the Order No is 'ABC':

Response

When using the Schedule – List API call, if any data has been found in Eyelit MES-M that meets the parameter values passed in the original payload, a JSON payload will be returned containing data in the following structure:

```
[
  {
    "id": 0,
    "suid": "string",
    "orderNo": "string",
    "batchNo": "string",
    "qty": 0,
    "sequence": 0,
    "description": "string",
    "estimatedTimePerUnit": 0,
    "itemVersion": "string",
    "isReleased": true,
    "batchSize": 0,
    "onHold": true,
    "holdReason": "string",
    "team": "string",
    "jobType": "string",
    "jobStatus": "string",
    "workflowVersion": "string",
    "salesOrderRef": "string",
    "userID": 0,
    "dateCompleted": "2024-07-09T20:47:22.355Z",
    "dueDate": "2024-07-09T20:47:22.355Z",
    "endDate": "2024-07-09T20:47:22.355Z",
    "startDate": "2024-07-09T20:47:22.355Z",
    "dateCreated": "2024-07-09T20:47:22.355Z",
    "jobPriority": "string",
    "customer": "string",
    "qtyStarted": 0,
    "qtyCompleted": 0,
    "qtyScrapped": 0,
    "recipe": "string",
    "bomVersion": "string",
    "testMode": "string",
    "uom": "string",
    "properties": [
      {
        "name": "string",
        "value": "string"
      }
    ],
    "serialNos": [
      "string"
    ],
    "notes": [
      "string"
    ]
  }
]
```


Sample Response

See below for sample use cases with examples of the JSON payload format returned.

No results were found that matched given parameters:


```
{

}
```


Single scheduled Job found that matches given parameters:

```
[
  {
    "id": 1234,
    "suid": "ERPON123",
    "orderNo": "ABC",
    "batchNo": "Batch1",
    "qty": 1,
    "sequence": 1,
    "description": "ERP Order",
    "estimatedTimePerUnit": 0,
    "itemVersion": "11",
    "isReleased": true,
    "batchSize": 10,
    "onHold": false,
    "holdReason": null,
    "team": null,
    "jobType": "Production",
    "jobStatus": "Not Started",
    "workflowVersion": "1.2",
    "salesOrderRef": "SO00123",
    "userID": 1543,
    "dateCompleted": "2024-07-09T20:47:22.355Z",
    "dueDate": "2024-07-09T20:47:22.355Z",
    "endDate": "2024-07-09T20:47:22.355Z",
    "startDate": "2024-07-09T20:47:22.355Z",
    "dateCreated": "2024-07-09T20:47:22.355Z",
    "jobPriority": "High",
    "customer": "customer 1",
    "qtyStarted": 0,
    "qtyCompleted": 0,
    "qtyScrapped": 0,
    "recipe": "Recipe 1",
    "bomVersion": "1.4",
    "testMode": "string",
    "uom": "EA",
    "properties":
    [
      {
        "name": "JobProp1",
        "value": "ABC"
      }
    ],
    "serialNos":
    [
      "SN12345"
    ],
    "notes":
    [
      "Job Note 1"
    ]
  }
]
```

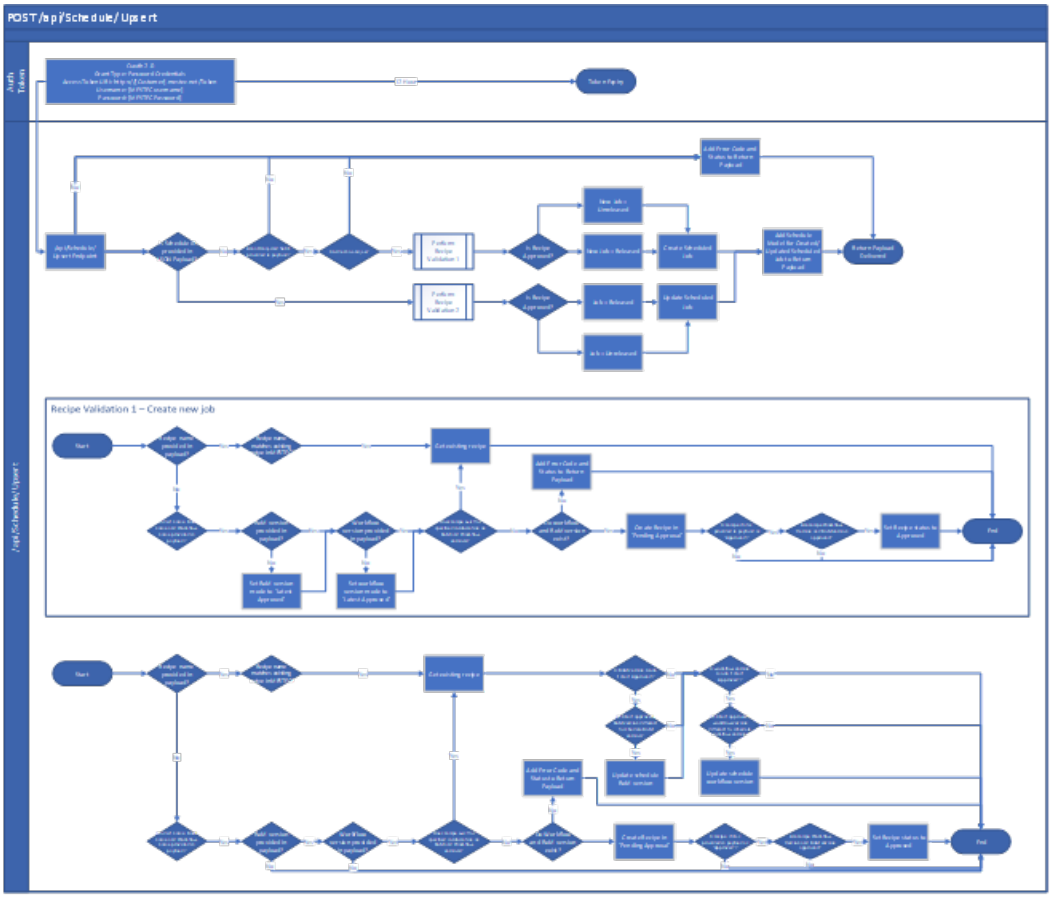

Upsert

When Schedule ID is provided, the Schedule – Upsert API call checks if a Scheduled Job with the given ID already exists. If it does, appropriate fields are updated in the appropriate Scheduled Job in Eyelit MES-M.

When the Schedule ID is omitted, a new Scheduled Job will be created.

See Figure 2 for logic.

Figure 2 - Logic within Schedule/Upsert API Call



Data Prerequisites

An active product, workflow version and BoM version must exist, at minimum, to be able to create a Scheduled Job in Eyelit MES-M. A recipe can be created either manually in Eyelit MES-M as a pre-requisite to the use of the Schedule – Upsert Public API endpoint, or can be created through use of the endpoint.

Request

Table 4 shows the method and endpoint required to make the API call to Upsert Scheduled Jobs.

Table 4 - Outbound Message Detail for Schedule - Upsert

Method	URL Structure	Endpoint
PUT	https://[environment].mestec.net	/api/Schedule/Upsert

The body of the payload should follow the format below.

```
{
  "id": 0,
  "suid": "string",
  "orderNo": "string",
  "batchNo": "string",
  "qty": 0,
  "description": "string",
  "estimatedTimePerUnit": 0,
  "itemVersion": "string",
  "isReleased": true,
  "batchSize": 0,
  "onHold": true,
  "holdReason": "string",
  "jobType": "string",
  "jobStatus": "string",
  "salesOrderRef": "string",
  "dueDate": "2024-11-25T09:08:20.499Z",
  "endDate": "2024-11-25T09:08:20.499Z",
  "startDate": "2024-11-25T09:08:20.499Z",
  "jobPriority": "string",
  "customer": "string",
  "uom": "string",
  "properties": [
    {
      "name": "string",
      "value": "string"
    }
  ],
  "serialNos": [
    "string"
  ],
  "notes": [
    "string"
  ],
  "recipe": {
    "recipeName": "string",
    "productName": "string",
    "bomName": "string",
    "workflowName": "string",
    "bomVersion": "string",
    "workflowVersion": "string",
    "recipeStatus": "string",
    "autoCreateRecipe": true
  },
  "testMode": "string"
}
```


See Table 5 for information on which fields are optional, the appropriate data types and the mappings to fields in Eyelit MES-M.

Table 5 - Parameter Information for Schedule - Upsert

Parameter Name	Data Mapping	Data Type	Mandatory		Case Sensitive	Match Type
			Create	Update		
id	schedule.id	NUMBER	N/A	Yes	N/A	Exact
suid	schedule.suid	VARCHAR2(200 BYTE)	No	No	N/A	N/A

orderNo	Schedule.orderno	VARCHAR2(50 BYTE)	No	No	N/A	N/A
batchNo	Schedule.batchno	VARCHAR2(50 BYTE)	No	No	N/A	N/A
qty	Schedule.qty	NUMBER	Yes	No	N/A	N/A
description	Schedule.description	VARCHAR2(500)	No	No	N/A	N/A
estimatedTimePerUnit	Schedule.estimatedtimeperunit	NUMBER	No	No	N/A	N/A
itemVersion	Schedule.itemversion	VARCHAR2(50)	No	No	N/A	N/A
isReleased	Schedule.isreleased	BOOL	Yes	No	N/A	N/A
batchSize	Schedule.batchsize	FLOAT	No	No	N/A	N/A
onHold	Schedule.onhold	BOOL	Yes	No	N/A	N/A
holdReason	Holdreason.name	VARCHAR2(50)	No1	No	No	Exact
jobType	Jobtype.name	VARCHAR2(20)	No	No	No	Exact
jobStatus	Jobstatus.name	VARCHAR2(20)	No	No	No	Exact
salesOrderRef	Schedule.salesorderref	VARCHAR2(50)	No	No	N/A	N/A
dueDate	Schedule.duedate	TIMESTAMP(6)	No	No	N/A	N/A
endDate	Schedule.enddate	TIMESTAMP(6)	No	No	N/A	N/A
startDate	Schedule.startdate	TIMESTAMP(6)	No	No	N/A	N/A
jobPriority	Jobpririty.name	VARCHAR2(100)	No	No	No	Exact
customer	Organisation.name	NVARCHAR2(255)	No	No	No	Exact
uom	Unitofmeasure.unitname	VARCHAR2(30)	No	No	No	Exact
testMode	TestMode.Name	VARCHAR2(200 BYTE)	Yes	--	No	Exact
properties		JSON Array	No	No	N/A	N/A
name	Property.name	VARCHAR2(50)	No	No	No	Exact
value	Scheduleprop.value	VARCHAR2(500)	No	No	N/A	N/A
serialNos	Jobserialno.serialno	VARCHAR2(50) (JSON Array)	No	No	N/A	N/A
notes	Tablepropertylog.comments	VARCHAR2(3000) (JSON Array)	No	No	N/A	N/A
recipe						
recipename	Recipe.name	VARCHAR2(200)	No	--	No	Exact

productname	Product.name	VARCHAR2(50)	No2	--	No	Exact
bomname	Bom.name	VARCHAR2(100)	No2	--	No	Exact
workflowname	Workflow.name	NVARCHAR2(200)	No2	--	No	Exact
bomversion	Bomversion.majorversion Bomversion.minorversion	VARCHAR2(10) X.X	No	--	N/A	Concatenated
workflowversion	Workflowversion.majorversion	VARCHAR2(10)	No	No	N/A	Concatenated
recipestatus	Workflowversion.minorversion Approvalstatus.name	X.X VARCHAR2(50) Pending Approval/Approved	No	--	No	Exact
autoCreateRecipe	Should Recipe be Auto Created	BOOL	Yes	N/A	N/A	N/A

Note: For any fields where the match type is ‘ Concatenated’, each element of the input will be compared to an Eyelit MES-M field.
“--" indicate field cannot be updated.

testMode must be one of the following:

- Production
- Pre-Production
- Virtual Test

Sample Request

See below for sample use cases with examples of the JSON payload format required.

To create a Schedule Job:

```
{
  "id": 1234,
  "suid": "ERPON123",
  "orderNo": "ABC123",
  "batchNo": "Batch1",
  "qty": 1,
  "description": "ERP Order",
  "estimatedTimePerUnit": 0,
  "itemVersion": "11",
  "isReleased": true,
  "batchSize": 10,
  "onHold": false,
  "holdReason": null,
  "team": null,
  "jobType": "Production",
  "salesOrderRef": "SO00123",
  "userID": 1543,
  "dueDate": "2024-07-09T20:47:22.355Z",
  "endDate": "2024-07-09T20:47:22.355Z",
  "startDate": "2024-07-09T20:47:22.355Z",
  "jobPriority": "High",
  "customer": "customer 1",
  "testMode": "Pre-Production",
  "uom": "EA",
  "serialNos":
  [
    "SN12345"
  ],
  "recipe": {
    "productName": "Product 123",
    "bomName": "Production BOM A",
    "workflowName": "Production Workflow A",
    "recipeStatus": "Approved"
  }
}
```


To update due date by ID:


```
{
  "id": 28,
  "duedate": "2024-07-12T09:20:08.914Z"
}
```


Response

When using the Schedule – Upsert API call, if a Scheduled Job has been created or updated, a JSON payload will be returned containing data in the following structure:

```
[
  {
    "id": 0,
    "suid": "string",
    "orderNo": "string",
    "batchNo": "string",
    "qty": 0,
    "sequence": 0,
    "description": "string",
    "estimatedTimePerUnit": 0,
    "itemVersion": "string",
    "isReleased": true,
    "batchSize": 0,
    "onHold": true,
    "holdReason": "string",
    "team": "string",
    "jobType": "string",
    "jobStatus": "string",
    "workflowVersion": "string",
    "salesOrderRef": "string",
    "userID": 0,
    "dateCompleted": "2024-07-09T20:47:22.355Z",
    "dueDate": "2024-07-09T20:47:22.355Z",
    "endDate": "2024-07-09T20:47:22.355Z",
    "startDate": "2024-07-09T20:47:22.355Z",
    "dateCreated": "2024-07-09T20:47:22.355Z",
    "jobPriority": "string",
    "customer": "string",
    "qtyStarted": 0,
    "qtyCompleted": 0,
    "qtyScrapped": 0,
    "recipe": "string",
    "bomVersion": "string",
    "testMode": "string",
    "uom": "string",
    "properties": [
      {
        "name": "string",
        "value": "string"
      }
    ],
    "serialNos": [
      "string"
    ],
    "notes": [
      "string"
    ]
  }
]
```


Sample Response

See below for sample use cases with examples of the JSON payload format returned.

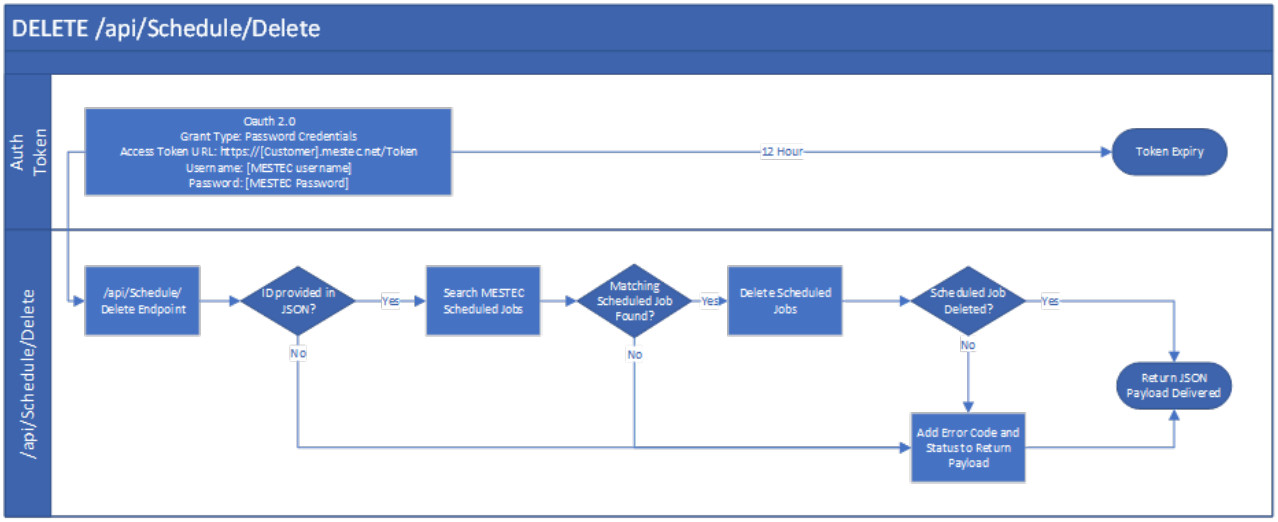
```
[
  {
    "id": 1234,
    "suid": "ERPON123",
    "orderNo": "ABC123",
    "batchNo": "Batch1",
    "qty": 1,
    "sequence": 1,
    "description": "ERP Order",
    "estimatedTimePerUnit": 0,
    "itemVersion": "11",
    "isReleased": true,
    "batchSize": 10,
    "onHold": false,
    "holdReason": null,
    "team": null,
    "jobType": "Production",
    "jobStatus": "Not Started",
    "workflowVersion": "1.2",
    "salesOrderRef": "SO00123",
    "userID": 1543,
    "dateCompleted": "2024-07-09T20:47:22.355Z",
    "dueDate": "2024-07-09T20:47:22.355Z",
    "endDate": "2024-07-09T20:47:22.355Z",
    "startDate": "2024-07-09T20:47:22.355Z",
    "dateCreated": "2024-07-09T20:47:22.355Z",
    "jobPriority": "High",
    "customer": "customer 1",
    "qtyStarted": 0,
    "qtyCompleted": 0,
    "qtyScrapped": 0,
    "recipe": "Recipe 1",
    "bomVersion": "1.4",
    "testMode": "Production",
    "uom": "EA",
    "properties":
    [
      {
        "name": "JobProp1",
        "value": "ABC"
      }
    ],
    "serialNos":
    [
      "SN12345"
    ],
    "notes":
    [
      "Job Note 1"
    ]
  }
]
```


Delete

The Schedule – Delete API call attempts to delete a Scheduled Job. A Scheduled Job can be deleted as long as it has not been started.

Figure 3 shows the logic used within the Schedule – Delete API call.

Figure 3 - Logic within Schedule/Delete API Call



Data Prerequisites

In order to delete a Scheduled Job it must exist in Eyelit MES-M and not have been started.

Request

Table 6 shows the method and endpoint required to make the API call to delete a Scheduled Job.

Table 6 - Outbound Message Detail for Schedule - Delete

Method	URL Structure	Endpoint
DELETE	https://[environment].mestec.net	/api/Schedule/Delete

The body of the payload should follow the format below.

```
{
  "id": 0
}
```

See Table 7 for information on which fields are optional, the appropriate data types and the mappings to fields in Eyelit MES-M.

Table 7 - Parameter Information for Schedule - Delete

Parameter Name	Data Mapping	Data Type	Mandatory	Case Sensitive	Match Type
ID	schedule.id	NUMBER	No	N/A	Exact

Sample Request

See below for sample use cases with examples of the JSON payload format required.

Delete by ID

```
{
  "id": 28
}
```

Response

When using the Schedule – Delete API call, if a Scheduled Job has been Deleted a 200 status response will be returned.