

# BoM Type

27/03/2026 2:17 pm GMT

Recipe BoM  
Tags: API API

## Versions

Versions this documentation is relevant for:

- 8.7.0+

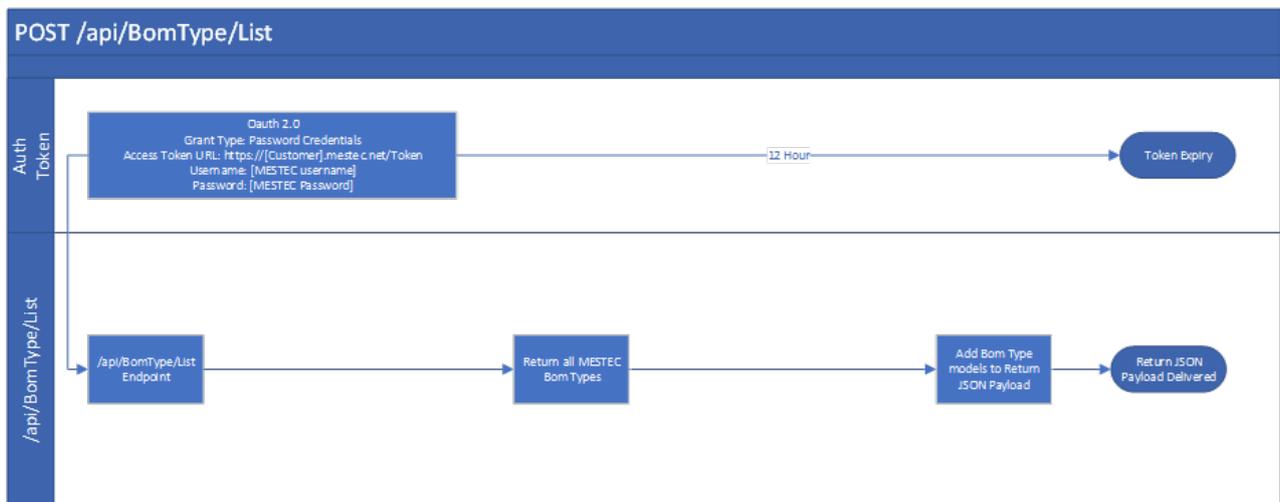
A BoM type is used to specify level of control required for a BoM. It is also used to specify that a BoM is job specific or not.

## List

The BoM – List API call reads existing BoM Types from Eyelit MES. The data can be filtered based on the parameters passed in the body/payload of the JSON packet giving the flexibility to search for one or multiple BoM Types.

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

Figure 1 - Logic within BoM Type/List API Call



## Data Prerequisites

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

## Request

Table 1 shows the method and endpoint required to make the API call to list Scheduled Jobs.

Table 1 - Outbound Message Detail for BoM Type - List

Method	URL Structure	Endpoint
--------	---------------	----------

POST	https://[environment].mestec.net /api/BomType/List
------	--

The body of the payload should follow the format below.

```
{  
}
```

Note that this transaction does not currently support any parameters.

## Sample Request

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

To list all BoM Types:

```
{  
}
```

## Response

When using the BoM Type – List API call, a JSON payload will be returned containing data in the following structure:

*JSON Structure for BoM Type List*

```
[  
  {  
    "id": 0,  
    "name": "string",  
    "description": "string",  
    "isJobSpecific": true,  
    "isControlled": true,  
    "requireApprovalSignature": true,  
    "approvalPermission": "string"  
  }  
]
```

## Sample Response

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

*JSON Sample for BoM Type List Response A*

```
[  
  {  
    "id": 10,  
    "name": "Bom Type 1",  
    "description": "ERP Bom Type",  
    "isJobSpecific": false,  
    "isControlled": true,  
    "requireApprovalSignature": true,  
    "approvalPermission": "Approved"  
  }  
]
```

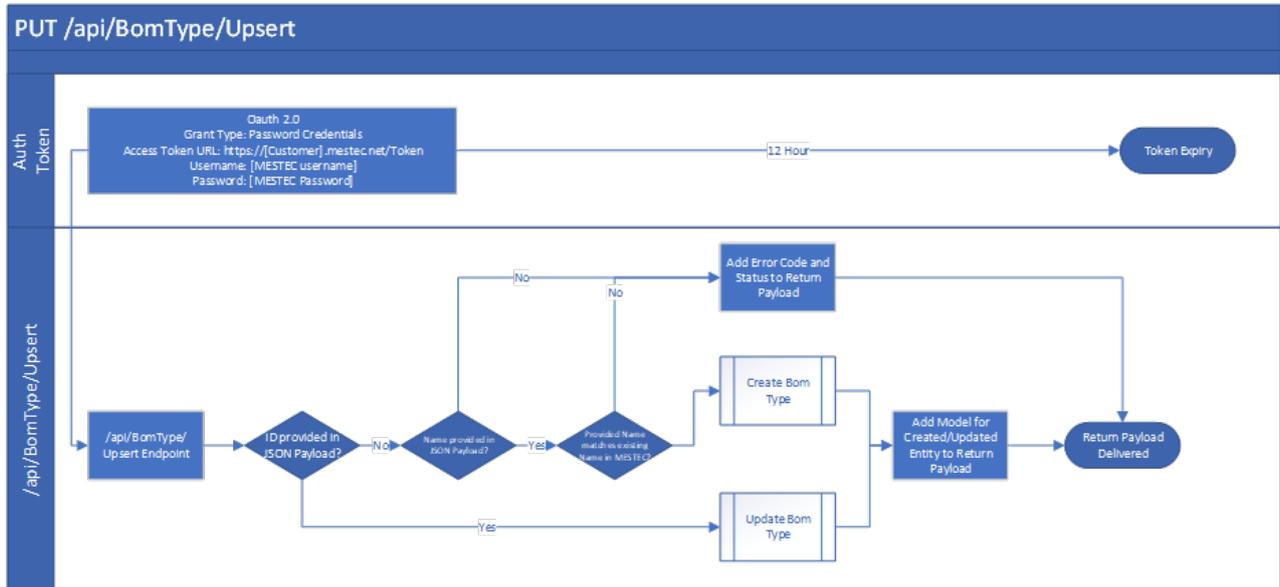
## Upsert

When BoM Type ID is provided, The BoM Type – Upsert API call checks if a BoM Type with the given ID already exists. If it

does, appropriate fields are updated in the matching BoM Type in Eyelit MES.

When the BoM Type ID is omitted, a new BoM Type will be created.

Figure 2 - Logic within BoM Type/Upsert API Call



## Data Prerequisites

No prerequisites for this transaction.

## Request

Table 2 shows the method and endpoint required to make the API call to Upsert BoM Type.

Table 2 - Outbound Message Detail for BoM Type - Upsert

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

The body of the payload should follow the format below.

```
{
  "id": 0,
  "name": "string",
  "description": "string",
  "isJobSpecific": true,
  "isControlled": true,
  "requireApprovalSignature": true,
  "approvalPermission": "string"
}
```

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

Table 3 - Parameter Information for BoM Type - Upsert

Parameter Name	Data Mapping	Data Type	Mandatory		Case Sensitive	Match Type
			Create	Update		
Id	Bomtype.id	NUMBER(10,0)	N/A	Yes	N/A	Exact
Name	Bomtype.name	VARCHAR2(50 BYTE)	Yes	No	N/A	N/A
Description	Bomtype.description	VARCHAR2(250 BYTE)	No	No	N/A	N/A
isJobSpecific	Bomtype.isjobspecific	BOOL	Yes	No	N/A	N/A
isControlled	Bomtype.iscontrolled	BOOL	Yes	No	N/A	N/A
requireApprovalSignature	Bomtype.requireapprovalsignature	BOOL	Yes	No	N/A	N/A
approvalPermission	Permission.permissionname	VARCHAR2(100 BYTE)	Yes	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 create a BoM Type.

```
{
  "name": "Bom Type 1",
  "description": "ERP Bom Type",
  "isJobSpecific": false,
  "isControlled": true,
  "requireApprovalSignature": true,
  "approvalPermission": "Bom Approval"
}
```

To update description by ID.

```
{
  "id": 10,
  "description": "Updated through API Call"
}
```

## Response

When using the BoM Type – Upsert API call, if a BoM Type has been Created or Updated a JSON payload will be returned containing data in the following structure:

```

{
  "id": 0,
  "name": "string",
  "description": "string",
  "isJobSpecific": true,
  "isControlled": true,
  "requireApprovalSignature": true,
  "approvalPermission": "string"
}

```

## Sample Response

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

```

{
  "id": 10,
  "name": "Bom Type 1",
  "description": "Updated through API Call",
  "isJobSpecific": false,
  "isControlled": true,
  "requireApprovalSignature": true,
  "approvalPermission": "Bom Approval"
}

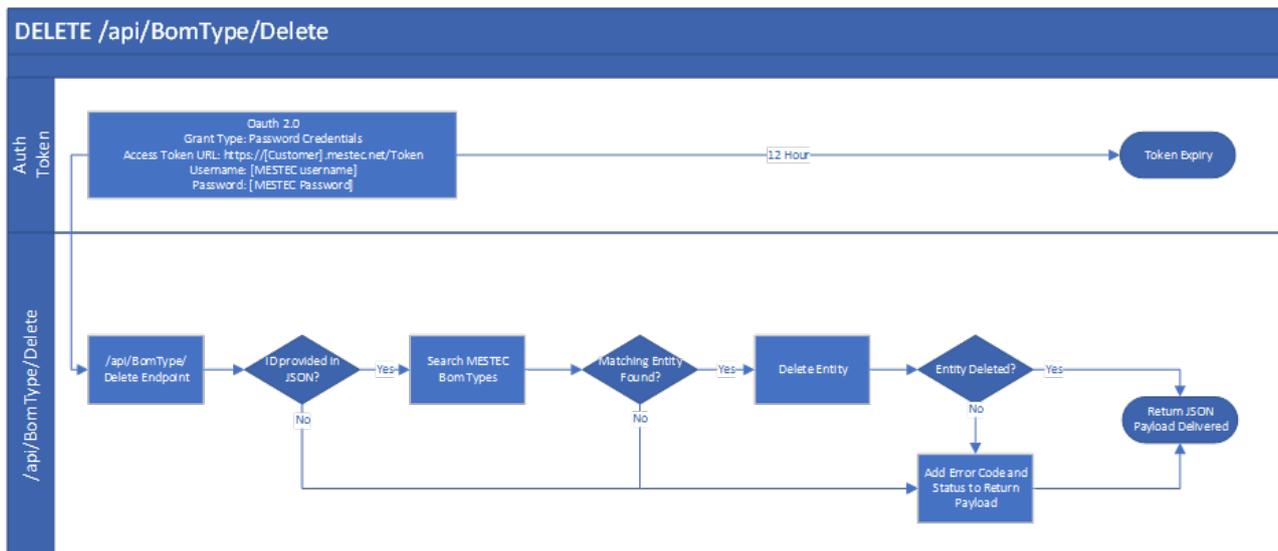
```

# Delete

The BoM Type – Delete API call attempts to delete a BoM Type. A BoM Type can be deleted as long as it has not been used.

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

Figure 3 - Logic within BoM Type/Delete API Call



## Data Prerequisites

In order to delete a BoM Type it must exist in Eyelit MES and not currently be in use.

## Request

Table 4 shows the method and endpoint required to make the API call to delete a BoM Type.

Table 4 - Outbound Message Detail for Schedule - Delete

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

The body of the payload should follow the format below.

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

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

Table 5 - Parameter Information for BoM Type - Delete

Parameter Name	Data Mapping	Data Type	Mandatory	Case Sensitive	Match Type
ID	BomType.id	NUMBER	No	N/A	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. See examples in Section 3.1.3.1 for use cases.

## Sample Request

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

Delete by ID

```
{  
  "id": 10  
}
```

## Response

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

---