

API Reference

iDimSS API

API Version: v1.0

INDEX

| | |
|----------------------------------|-----------|
| 1. AUTHENTICATE | 4 |
| 1.1 POST /api/Authenticate/login | 4 |
| 2. CAPTURE | 5 |
| 2.1 POST /api/Capture | 5 |
| 3. DIMENSIONER | 7 |
| 3.1 GET /api/Dimensioner | 7 |
| 3.2 PUT /api/Dimensioner | 7 |
| 3.3 POST /api/Dimensioner | 10 |
| 3.4 DELETE /api/Dimensioner | 13 |
| 3.5 GET /api/Dimensioner/{id} | 14 |
| 4. FORKLIFT | 16 |
| 4.1 GET /api/Forklift | 16 |
| 4.2 PUT /api/Forklift | 16 |
| 4.3 POST /api/Forklift | 18 |
| 4.4 DELETE /api/Forklift | 19 |
| 4.5 GET /api/Forklift/{id} | 19 |
| 5. MEASUREMENT | 21 |
| 5.1 GET /api/M Measurement | 21 |
| 5.2 GET /api/M Measurement/{id} | 22 |
| 6. SCALE | 23 |
| 6.1 GET /api/Scale | 23 |
| 6.2 PUT /api/Scale | 23 |
| 6.3 POST /api/Scale | 25 |
| 6.4 DELETE /api/Scale | 26 |
| 6.5 GET /api/Scale/{id} | 27 |

Security and Authentication

SECURITY SCHEMES

| KEY | TYPE | DESCRIPTION |
|--------|--------|--|
| oauth2 | apiKey | Standard authorization Header using Bearer scheme. Example: "Bearer {token}" |

API

1. AUTHENTICATE

1.1 POST /api/Authenticate/login

Login to obtain a token for use with the APIs.

REQUEST

REQUEST BODY - application/json

```
{
  password* string
  userName* string
}
```

REQUEST BODY - text/json

```
{
  password* string
  userName* string
}
```

REQUEST BODY - application/*+json

```
{
  password* string
  userName* string
}
```

RESPONSE

STATUS CODE - 200: Returns the Jwt token with the expiration date.

RESPONSE MODEL - application/json

```
{
  expiration string
  token      string
}
```

STATUS CODE - 400: If the login model was not valid or was null.

STATUS CODE - 401: If provided credentials were not accepted.

2. CAPTURE

2.1 POST /api/Capture

Triggers a capture operation using the details in the **CaptureRequestModel**.

Sample request: `POST /Capture{"DimensionerName": "TEST","ProNumber": "PRO368127368236","UserField1": "user value","UserField2": "user value","UserField3": "user value",}`

REQUEST

REQUEST BODY - application/json

```
{
  dimensionerName* string 0 to 50 chars
                                The name of the dimensioner used to perform the capture operation.
  proNumber*          string 0 to 100 chars
                                The 'ProNumber' or shipping manifest identifier.
  weight              number An optional externally provided weight value.
  weightUnits         string 0 to 15 chars
                                An optional externally provided weight unit of measure.
  userField1          string 0 to 255 chars
                                An optional value for user defined field one.
  userField2          string 0 to 255 chars
                                An optional value for user defined field two.
  userField3          string 0 to 255 chars
                                An optional value for user defined field three.
  forkliftId          string 0 to 30 chars
                                An optional value for the Forklift Id. Required to trigger a stop and go capture process.
}
```

REQUEST BODY - text/json

```
{
  dimensionerName* string 0 to 50 chars
                                The name of the dimensioner used to perform the capture operation.
  proNumber*          string 0 to 100 chars
                                The 'ProNumber' or shipping manifest identifier.
  weight              number An optional externally provided weight value.
  weightUnits         string 0 to 15 chars
                                An optional externally provided weight unit of measure.
  userField1          string 0 to 255 chars
                                An optional value for user defined field one.
  userField2          string 0 to 255 chars
                                An optional value for user defined field two.
  userField3          string 0 to 255 chars
                                An optional value for user defined field three.
  forkliftId          string 0 to 30 chars
                                An optional value for the Forklift Id. Required to trigger a stop and go capture process.
}
```

REQUEST BODY - application/*+json

```
{
  dimensionerName* string 0 to 50 chars
                                The name of the dimensioner used to perform the capture operation.
  proNumber*          string 0 to 100 chars
}
```

| | | |
|-------------|--------|---|
| | | The 'ProNumber' or shipping manifest identifier. |
| weight | number | An optional externally provided weight value. |
| weightUnits | string | 0 to 15 chars |
| | | An optional externally provided weight unit of measure. |
| userField1 | string | 0 to 255 chars |
| | | An optional value for user defined field one. |
| userField2 | string | 0 to 255 chars |
| | | An optional value for user defined field two. |
| userField3 | string | 0 to 255 chars |
| | | An optional value for user defined field three. |
| forkliftId | string | 0 to 30 chars |
| | | An optional value for the Forklift Id. Required to trigger a stop and go capture process. |

}

RESPONSE

STATUS CODE - 201: Returns the capture response.

RESPONSE MODEL - application/json

```
{
  captureDate      string  The date and time of the capture operation.
  captureId        integer  The capture id assigned by the dimensioner.
  dimUnit          string  The dimension unit of measure.
  displayWeight    string  The formatted weight of the object.
  forkliftId       string  The forklift id used to trigger a stop and go capture process.
  height           number  The height of the measured object.
  images [{
    Array of object:
      name          string  The image name.
      base64Image   string  The image bytes encoded in Base64.
  }]
  isOversized      boolean  A flag that indicates that the measured object is oversized
                        based on iDimension legal-for-trade approvals.
  isUndersized     boolean  A flag that indicates that the measured object is undersized
                        based on iDimension legal-for-trade approvals.
  length           number  The length of the measured object.
  proNumber        string  The ProNumber or shipping manifest identifier.
  userField1       string  An optional value for user defined field one.
  userField2       string  An optional value for user defined field two.
  userField3       string  An optional value for user defined field three.
  userFieldLabel1  string  The label for user defined field number 1.
  userFieldLabel2  string  The label for user defined field number 2.
  userFieldLabel3  string  The label for user defined field number 3.
  volume           number  READ-ONLY
                        The volume of the measured object.
  weight           number  The weight of the object.
  weightUnit       string  The weight unit of measure.
  width           number  The width of the measured object.
}
```

STATUS CODE - 400: If the CaptureRequestModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

3. DIMENSIONER

3.1 GET /api/Dimensioner

Returns a list of all Dimensioners.

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Returns the list of dimensioners.

RESPONSE MODEL - application/json

[{

Array of object:

| | | |
|-------------------|---------|--|
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |
| }] | | |

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

3.2 PUT /api/Dimensioner

Update an existing dimensioner using the data in the dimensioner model. (Auth)

Sample request: PUT /Dimensioner{"Id": 1,"Name": "EastDock","DimensionerType": "LTL","IpAddress": "192.168.0.40","AnnotateImages": true,"CameraNames": "QVRemoteImage1,QVRemoteImage2,QVRemoteImage3,QVRemoteImage4,QVRemoteImage5,QVRemoteImage6","CaptureDefinition": "QVDemo","CompositeImages": true,"HasAttachedScale": false,"HasExternalScale": false,"HotspotName": "HS1","ProcessImages": "QVRemoteCameraImage1","AssociatedScaleId": 4,}

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|----------------------------------|
| id | int32 | | the Id of the dimensioner model. |

REQUEST BODY - application/json

| | | |
|-------------------|---------|---|
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

REQUEST BODY - text/json

| | | |
|-------------------|---------|---|
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |

| | | | |
|-------------------|---------|-----------------|--|
| captureDefinition | string | 0 to 50 chars | The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars | The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars | A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars | The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars | PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars | The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars | A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

}

REQUEST BODY - application/*+json

| | | | |
|-------------------|---------|-----------------|--|
| annotateImages | boolean | | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars | A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars | The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars | The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars | A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars | The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars | PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars | The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars | A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

}

RESPONSE

STATUS CODE - 200: Returns the DimensionerModel.

RESPONSE MODEL - application/json

| | | |
|-------------------|---------|---|
| { | | |
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5]))\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |
| } | | |

STATUS CODE - 400: If the DimensionerModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

3.3 POST /api/Dimensioner

Add a new dimensioner using the data in the dimensioner model. (Auth)

Sample request: POST /Dimensioner{"Name": "EastDock","DimensionerType": "LTL","IpAddress": "192.168.0.40","AnnotateImages": true,"CameraNames": "QVRemoteImage1,QVRemoteImage2,QVRemoteImage3,QVRemoteImage4,QVRemoteImage5,QVRemoteImage6,QVRemoteImage7,QVRemoteImage8,QVRemoteImage9,QVRemoteImage10,QVRemoteImage11,QVRemoteImage12,QVRemoteImage13,QVRemoteImage14,QVRemoteImage15,QVRemoteImage16,QVRemoteImage17,QVRemoteImage18,QVRemoteImage19,QVRemoteImage20,QVRemoteImage21,QVRemoteImage22,QVRemoteImage23,QVRemoteImage24,QVRemoteImage25,QVRemoteImage26,QVRemoteImage27,QVRemoteImage28,QVRemoteImage29,QVRemoteImage30,QVRemoteImage31,QVRemoteImage32,QVRemoteImage33,QVRemoteImage34,QVRemoteImage35,QVRemoteImage36,QVRemoteImage37,QVRemoteImage38,QVRemoteImage39,QVRemoteImage40,QVRemoteImage41,QVRemoteImage42,QVRemoteImage43,QVRemoteImage44,QVRemoteImage45,QVRemoteImage46,QVRemoteImage47,QVRemoteImage48,QVRemoteImage49,QVRemoteImage50,QVRemoteImage51,QVRemoteImage52,QVRemoteImage53,QVRemoteImage54,QVRemoteImage55,QVRemoteImage56,QVRemoteImage57,QVRemoteImage58,QVRemoteImage59,QVRemoteImage60,QVRemoteImage61,QVRemoteImage62,QVRemoteImage63,QVRemoteImage64,QVRemoteImage65,QVRemoteImage66,QVRemoteImage67,QVRemoteImage68,QVRemoteImage69,QVRemoteImage70,QVRemoteImage71,QVRemoteImage72,QVRemoteImage73,QVRemoteImage74,QVRemoteImage75,QVRemoteImage76,QVRemoteImage77,QVRemoteImage78,QVRemoteImage79,QVRemoteImage80,QVRemoteImage81,QVRemoteImage82,QVRemoteImage83,QVRemoteImage84,QVRemoteImage85,QVRemoteImage86,QVRemoteImage87,QVRemoteImage88,QVRemoteImage89,QVRemoteImage90,QVRemoteImage91,QVRemoteImage92,QVRemoteImage93,QVRemoteImage94,QVRemoteImage95,QVRemoteImage96,QVRemoteImage97,QVRemoteImage98,QVRemoteImage99,QVRemoteImage100","CaptureDefinition": "QVDemo","CompositeImages": true,"HasAttachedScale": false,"HasExternalScale": false,"HotspotName": "HS1","ProcessImages": "QVRemoteCameraImage1","AssociatedScaleId": 4,}

REQUEST

REQUEST BODY - application/json

| | | |
|-------------------|---------|---|
| { | | |
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |
| } | | |

REQUEST BODY - text/json

| | | |
|-------------------|---------|---|
| { | | |
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ |

| | | |
|----------------------|---------------|--|
| name* | string | The IP address of the dimensioner. 0 to 50 chars |
| processImages | string | The friendly name of the dimensioner. 0 to 1024 chars |
| | | A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

}

REQUEST BODY - application/*+json

```
{
  annotateImages      boolean  A flag that indicates whether to annotate each processed image with the measurement data.
  associatedScaleId   integer   The unique id of a scale associated with the dimensioner. This scale will be interrogated
                                for weight data during the measurement capture process.
  cameraNames        string    0 to 1024 chars
                                A comma delimited list of camera names. The system will ensure that each camera in this list
                                has been interrogated as part of the measurement process.
  captureDefinition  string    0 to 50 chars
                                The capture definition used by the dimensioner to perform the measurement.
  compositeImages     boolean  A flag that determines whether the measurement process composites/combines all
                                captured images into a single image.
  dimensionerType     string    0 to 100 chars
                                The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS
  exportImages        string    0 to 1024 chars
                                A comma delimited list of camera names. The images from the cameras
                                will be exported via FTP and/or SMB.
  hasAttachedScale    boolean  A flag that indicates that the dimensioner has a physically connected scale.
  hasExternalScale    boolean  A flag that indicates that any weight data included with the measurement will be
                                included as part of the capture operation.
  hotspotName         string    0 to 50 chars
                                The hotspot name for the dimensioner. Not all dimensioners use this value.
  id                  integer   The unique id of the dimensioner.
  ipAddress*          string    0 to 1024 chars
                                PATTERN: ^((([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|
                                25[0-5])\.){3}([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|
                                25[0-5]))$
                                The IP address of the dimensioner.
  name*              string    0 to 50 chars
                                The friendly name of the dimensioner.
  processImages       string    0 to 1024 chars
                                A comma delimited list of camera names. The images from the cameras
                                will be composited and/or annotated and included as part of the measurement.
}
```

RESPONSE

STATUS CODE - 201: Returns the DimensionerModel.

RESPONSE MODEL - application/json

```
{
  annotateImages      boolean  A flag that indicates whether to annotate each processed image with the measurement data.
  associatedScaleId   integer   The unique id of a scale associated with the dimensioner. This scale will be interrogated
                                for weight data during the measurement capture process.
  cameraNames        string    0 to 1024 chars
                                A comma delimited list of camera names. The system will ensure that each camera in this list
                                has been interrogated as part of the measurement process.
  captureDefinition  string    0 to 50 chars
                                The capture definition used by the dimensioner to perform the measurement.
  compositeImages     boolean  A flag that determines whether the measurement process composites/combines all
                                captured images into a single image.
  dimensionerType     string    0 to 100 chars
                                The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS
  exportImages        string    0 to 1024 chars
```

| | | |
|------------------|---------|--|
| hasAttachedScale | boolean | A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasExternalScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hotspotName | string | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| id | integer | 0 to 50 chars |
| ipAddress* | string | The hotspot name for the dimensioner. Not all dimensioners use this value. |
| | | The unique id of the dimensioner. |
| | | 0 to 1024 chars |
| | | PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ |
| | | The IP address of the dimensioner. |
| name* | string | 0 to 50 chars |
| | | The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars |
| | | A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

}

STATUS CODE - 400: If the DimensionerModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

3.4 DELETE /api/Dimensioner

(Auth)

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-------------|
| id | int32 | | |

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

| | | |
|-------------------|---------|--|
| { | | |
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars |
| | | A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars |
| | | The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars |
| | | The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars |
| | | A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |

| | | |
|------------------|---------|---|
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |
| } | | |

STATUS CODE - 401: Unauthorized

STATUS CODE - 403: Forbidden

3.5 GET /api/Dimensioner/{id}

Returns a Dimensioner.

REQUEST

PATH PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-----------------------------|
| *id | int32 | | The dimensioner identifier. |

RESPONSE

STATUS CODE - 200: Returns the dimensioner.

RESPONSE MODEL - application/json

| | | |
|-------------------|---------|---|
| { | | |
| annotateImages | boolean | A flag that indicates whether to annotate each processed image with the measurement data. |
| associatedScaleId | integer | The unique id of a scale associated with the dimensioner. This scale will be interrogated for weight data during the measurement capture process. |
| cameraNames | string | 0 to 1024 chars A comma delimited list of camera names. The system will ensure that each camera in this list has been interrogated as part of the measurement process. |
| captureDefinition | string | 0 to 50 chars The capture definition used by the dimensioner to perform the measurement. |
| compositeImages | boolean | A flag that determines whether the measurement process composites/combines all captured images into a single image. |
| dimensionerType | string | 0 to 100 chars The type of dimensioner. Acceptable values are: LTL, PWD, IDIMPLUS |
| exportImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be exported via FTP and/or SMB. |
| hasAttachedScale | boolean | A flag that indicates that the dimensioner has a physically connected scale. |
| hasExternalScale | boolean | A flag that indicates that any weight data included with the measurement will be included as part of the capture operation. |
| hotspotName | string | 0 to 50 chars The hotspot name for the dimensioner. Not all dimensioners use this value. |

| | | |
|---------------|---------|--|
| id | integer | The unique id of the dimensioner. |
| ipAddress* | string | 0 to 1024 chars PATTERN: ^(([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\.){3}([0-9] [1-9][0-9] 1[0-9]{2} 2[0-4][0-9] 25[0-5])\$ The IP address of the dimensioner. |
| name* | string | 0 to 50 chars The friendly name of the dimensioner. |
| processImages | string | 0 to 1024 chars A comma delimited list of camera names. The images from the cameras will be composited and/or annotated and included as part of the measurement. |

}

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

4. FORKLIFT

4.1 GET /api/Forklift

Returns a list of all Forklifts. (Auth)

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Returns the list of Forklifts.

RESPONSE MODEL - application/json

```
[{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer The x-offset for the forklift markers in millimeters.
  yOffset       integer between -2147483648 and 0
                  The y-offset for the forklift markers in millimeters.
  zOffset       integer between 0 and 2147483647
                  The z-offset for the forklift markers in millimeters.
}]
```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: Forbidden

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

4.2 PUT /api/Forklift

Update an existing Forklift using the data in the Forklift model. (Auth)

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-------------------------------|
| id | int32 | | the Id of the Forklift model. |

REQUEST BODY - application/json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer between 0 and 2147483647
```


| | | |
|---------|---------|---|
| | | The distance of the markers from the centroid installed on the forklift in millimeters. |
| xOffset | integer | The x-offset for the forklift markers in millimeters. |
| yOffset | integer | between -2147483648 and 0 |
| | | The y-offset for the forklift markers in millimeters. |
| zOffset | integer | between 0 and 2147483647 |
| | | The z-offset for the forklift markers in millimeters. |

```

}
```

REQUEST BODY - text/json

```

{
  forkliftId*   string  0 to 30 chars
                The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                The z-offset for the forklift markers in millimeters.
}
```

REQUEST BODY - application/*+json

```

{
  forkliftId*   string  0 to 30 chars
                The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                The z-offset for the forklift markers in millimeters.
}
```

RESPONSE

STATUS CODE - 200: Returns the ForkliftModel.

RESPONSE MODEL - application/json

```

{
  forkliftId*   string  0 to 30 chars
                The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                The z-offset for the forklift markers in millimeters.
}
```

STATUS CODE - 400: If the ForkliftModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

4.3 POST /api/Forklift

Add a new Forklift using the data in the Forklift model. (Auth)

REQUEST

REQUEST BODY - application/json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                  The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                  The z-offset for the forklift markers in millimeters.
}
```

REQUEST BODY - text/json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                  The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                  The z-offset for the forklift markers in millimeters.
}
```

REQUEST BODY - application/*+json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                  The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                  The z-offset for the forklift markers in millimeters.
}
```

RESPONSE

STATUS CODE - 201: Returns the ForkliftModel.

RESPONSE MODEL - application/json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
}
```

```

    xOffset      integer  The x-offset for the forklift markers in millimeters.
    yOffset      integer  between -2147483648 and 0
                                The y-offset for the forklift markers in millimeters.
    zOffset      integer  between 0 and 2147483647
                                The z-offset for the forklift markers in millimeters.
}

```

STATUS CODE - 400: If the ForkliftModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

4.4 DELETE /api/Forklift

(Auth)

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-------------|
| id | int32 | | |

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```

{
  forkliftId*   string  0 to 30 chars
                        The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                        The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                        The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                        The z-offset for the forklift markers in millimeters.
}

```

STATUS CODE - 401: Unauthorized

STATUS CODE - 403: Forbidden

4.5 GET /api/Forklift/{id}

Returns a Forklift. (Auth)

REQUEST

PATH PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|------|---------|-------------|
|------|------|---------|-------------|

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|--------------------------|
| *id | int32 | | The Forklift identifier. |

RESPONSE

STATUS CODE - 200: Returns the Forklift.

RESPONSE MODEL - application/json

```
{
  forkliftId*   string  0 to 30 chars
                  The user provided forklift identifier.
  id            integer
  markerDistance integer  between 0 and 2147483647
                  The distance of the markers from the centroid installed on the forklift in millimeters.
  xOffset       integer  The x-offset for the forklift markers in millimeters.
  yOffset       integer  between -2147483648 and 0
                  The y-offset for the forklift markers in millimeters.
  zOffset       integer  between 0 and 2147483647
                  The z-offset for the forklift markers in millimeters.
}
```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

5. MEASUREMENT

5.1 GET /api/Measurement

Returns a list of all Measurements.

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Returns the list of measurements.

RESPONSE MODEL - application/json

```
[ {  
  Array of object:  
    captureDate      string  The date and time of the capture operation.  
    captureId        integer The capture id assigned by the dimensioner.  
    dimensionerName   string  The name of the dimensioner used to perform the measurement.  
    dimUnit          string  The dimension unit of measure.  
    displayWeight     string  The formatted weight.  
    forkliftId        string  The forklift Id (if stop and go was used).  
    height            number  The height of the measured object.  
    id                integer The unique id of the measurement.  
    images [ {  
      Array of object:  
        name          string  The image name.  
        base64Image    string  The image bytes encoded in Base64.  
      } ]  
    isOversized       boolean A flag that indicates that the measured object is oversized  
                        based on iDimension legal-for-trade approvals.  
    isUndersized      boolean A flag that indicates that the measured object is undersized  
                        based on iDimension legal-for-trade approvals.  
    length            number  The length of the measured object.  
    proNumber         string  The ProNumber or shipping manifest identifier.  
    userField1        string  The value of user defined field number 1.  
    userField2        string  The value of user defined field number 2.  
    userField3        string  The value of user defined field number 2.  
    userFieldLabel1   string  The label for user defined field number 1.  
    userFieldLabel2   string  The label for user defined field number 2.  
    userFieldLabel3   string  The label for user defined field number 3.  
    volume            number  READ-ONLY  
                        The volume of the measured object.  
    weight            number  The weight of the measured object.  
    weightUnit        string  The weight unit of measure.  
    width             number  The width of the object.  
  } ]
```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

5.2 GET /api/Measurement/{id}

Returns a Measurement.

REQUEST

PATH PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-----------------------------|
| *id | int32 | | The measurement identifier. |

RESPONSE

STATUS CODE - 200: Returns the measurement.

RESPONSE MODEL - application/json

```
{
  captureDate      string  The date and time of the capture operation.
  captureId        integer  The capture id assigned by the dimensioner.
  dimensionerName  string  The name of the dimensioner used to perform the measurement.
  dimUnit          string  The dimension unit of measure.
  displayWeight    string  The formatted weight.
  forkliftId       string  The forklift Id (if stop and go was used).
  height           number  The height of the measured object.
  id               integer  The unique id of the measurement.
  images [{
    Array of object:
      name          string  The image name.
      base64Image   string  The image bytes encoded in Base64.
    }]
  isOversized      boolean  A flag that indicates that the measured object is oversized
                        based on iDimension legal-for-trade approvals.
  isUndersized     boolean  A flag that indicates that the measured object is undersized
                        based on iDimension legal-for-trade approvals.
  length           number  The length of the measured object.
  proNumber        string  The ProNumber or shipping manifest identifier.
  userField1       string  The value of user defined field number 1.
  userField2       string  The value of user defined field number 2.
  userField3       string  The value of user defined field number 2.
  userFieldLabel1  string  The label for user defined field number 1.
  userFieldLabel2  string  The label for user defined field number 2.
  userFieldLabel3  string  The label for user defined field number 3.
  volume           number  READ-ONLY
                        The volume of the measured object.
  weight           number  The weight of the measured object.
  weightUnit       string  The weight unit of measure.
  width            number  The width of the object.
}
```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

6. SCALE

6.1 GET /api/Scale

Returns a list of all Scales. (Auth)

REQUEST

No request parameters

RESPONSE

STATUS CODE - 200: Returns the list of scales.

RESPONSE MODEL - application/json

```
[ {
  id          integer  The unique id of the scale.
  ipAddress*  string   0 to 1024 chars
                  PATTERN: ^([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                  The IP address of the scale.
  name*       string   0 to 50 chars
                  The name of the scale.
  scaleType*  string   0 to 100 chars
                  The scale type.
  tcpPort     integer  between 1 and 65535
                  The tcp port the scale uses to accept connections.
}
```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: Forbidden

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.

6.2 PUT /api/Scale

Update an existing scale using the data in the scale model. (Auth)

Sample request: PUT /Scale{"Id": 1,"Name": "EastScale","ScaleType": "880","IpAddress": "192.168.0.40","TcpPort": 10001}

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|----------------------------|
| id | int32 | | the Id of the scale model. |

REQUEST BODY - application/json

```
{
```

```

id            integer  The unique id of the scale.
ipAddress*   string   0 to 1024 chars
                  PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                  ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                  The IP address of the scale.
name*        string   0 to 50 chars
                  The name of the scale.
scaleType*   string   0 to 100 chars
                  The scale type.
tcpPort       integer  between 1 and 65535
                  The tcp port the scale uses to accept connections.
}

```

REQUEST BODY - text/json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                    PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                    ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                    The IP address of the scale.
  name*        string   0 to 50 chars
                    The name of the scale.
  scaleType*   string   0 to 100 chars
                    The scale type.
  tcpPort       integer  between 1 and 65535
                    The tcp port the scale uses to accept connections.
}

```

REQUEST BODY - application/*+json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                    PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                    ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                    The IP address of the scale.
  name*        string   0 to 50 chars
                    The name of the scale.
  scaleType*   string   0 to 100 chars
                    The scale type.
  tcpPort       integer  between 1 and 65535
                    The tcp port the scale uses to accept connections.
}

```

RESPONSE

STATUS CODE - 200: Returns the ScaleModel.

RESPONSE MODEL - application/json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                    PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                    ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                    The IP address of the scale.
  name*        string   0 to 50 chars
                    The name of the scale.
  scaleType*   string   0 to 100 chars
                    The scale type.
  tcpPort       integer  between 1 and 65535

```


The tcp port the scale uses to accept connections.

}

STATUS CODE - 400: If the ScaleModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

6.3 POST /api/Scale

Add a new scale using the data in the scale model. (Auth)

Sample request: POST /Scale{"Name": "EastScale","ScaleType": "880","IpAddress": "192.168.0.40","TcpPort": 10001}

REQUEST

REQUEST BODY - application/json

```
{
  id          integer  The unique id of the scale.
  ipAddress*  string   0 to 1024 chars
                  PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                  ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                  The IP address of the scale.
  name*       string   0 to 50 chars
                  The name of the scale.
  scaleType*  string   0 to 100 chars
                  The scale type.
  tcpPort     integer  between 1 and 65535
                  The tcp port the scale uses to accept connections.
}
```

REQUEST BODY - text/json

```
{
  id          integer  The unique id of the scale.
  ipAddress*  string   0 to 1024 chars
                  PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                  ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                  The IP address of the scale.
  name*       string   0 to 50 chars
                  The name of the scale.
  scaleType*  string   0 to 100 chars
                  The scale type.
  tcpPort     integer  between 1 and 65535
                  The tcp port the scale uses to accept connections.
}
```

REQUEST BODY - application/*+json

```
{
  id          integer  The unique id of the scale.
  ipAddress*  string   0 to 1024 chars
                  PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                  ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                  The IP address of the scale.
  name*       string   0 to 50 chars
                  The name of the scale.
}
```

```

scaleType* string 0 to 100 chars
                The scale type.
tcpPort      integer between 1 and 65535
                The tcp port the scale uses to accept connections.
}

```

RESPONSE

STATUS CODE - 201: Returns the ScaleModel.

RESPONSE MODEL - application/json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                  PATTERN: ^([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]))$
                  The IP address of the scale.
  name*        string   0 to 50 chars
                  The name of the scale.
  scaleType*   string   0 to 100 chars
                  The scale type.
  tcpPort      integer  between 1 and 65535
                  The tcp port the scale uses to accept connections.
}

```

STATUS CODE - 400: If the ScaleModel was not valid or was null.

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 500: If something went wrong.

6.4 DELETE /api/Scale

(Auth)

REQUEST

QUERY PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-------------|
| id | int32 | | |

RESPONSE

STATUS CODE - 200: Success

RESPONSE MODEL - application/json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                  PATTERN: ^([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]))$
                  The IP address of the scale.
  name*        string   0 to 50 chars
                  The name of the scale.
}

```

```

    scaleType* string 0 to 100 chars
                        The scale type.
    tcpPort    integer between 1 and 65535
                        The tcp port the scale uses to accept connections.
}

```

STATUS CODE - 401: Unauthorized

STATUS CODE - 403: Forbidden

6.5 GET /api/Scale/{id}

Returns a Scale. (Auth)

REQUEST

PATH PARAMETERS

| NAME | TYPE | EXAMPLE | DESCRIPTION |
|------|-------|---------|-----------------------|
| *id | int32 | | The scale identifier. |

RESPONSE

STATUS CODE - 200: Returns the scale.

RESPONSE MODEL - application/json

```

{
  id            integer  The unique id of the scale.
  ipAddress*   string   0 to 1024 chars
                        PATTERN: ^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\.){3}
                        ([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$
                        The IP address of the scale.
  name*        string   0 to 50 chars
                        The name of the scale.
  scaleType*   string   0 to 100 chars
  tcpPort      integer  between 1 and 65535
                        The scale type.
                        The tcp port the scale uses to accept connections.
}

```

STATUS CODE - 401: If access is not authorized.

STATUS CODE - 403: If access is forbidden.

STATUS CODE - 404: If the item was not found.

STATUS CODE - 500: If something went wrong.