



GETTING STARTED GUIDE

# SERVICE PRO SERVICE PRO API

---

VERSION 10.29.8





## About MSI

---

MSI helps companies to improve the efficiency and effectiveness of their field workforce. Focused on the enterprise, scheduling, and the mobile worker, our applications include Mobile Field Service, Mobile Inspection, and Field Service Management Software.

## Contact us

office: (262) 241-7800

fax: (414) 446-8081

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

## Copyright

Copyright MSI 2023. All rights reserved.

Last updated 9/11/2023

# Contents

---

<b>Introduction to Service Pro API</b> .....	5
<b>Guidelines and requirements</b> .....	6
Supported API call methods.....	6
Service Pro API URL.....	6
Rate limits.....	6
CPU time limits.....	6
Retries.....	6
DateTime format requirements.....	7
CorrelationId and CorrelationGroupId headers.....	7
<b>Authentication</b> .....	8
<b>Get data</b> .....	9
<b>Get order data</b> .....	10
Get an order using the order number.....	10
Get an order using the order ID.....	12
Include deferred data for an order.....	13
<b>Get attachments</b> .....	15
Download an order note attachment.....	15
Download an inspection result.....	17
<b>Get Service Pro downtime information</b> .....	19
<b>About VO classes</b> .....	21
<b>Select and expand using OData</b> .....	22
Get order segments and order lines.....	22
<b>Create records</b> .....	24
<b>Create orders</b> .....	25
Create an order for a specific customer.....	26
Create an order for a specific piece of equipment.....	27
Create an order using a single POST request.....	28
<b>Create an order note</b> .....	30
<b>Create order lines</b> .....	31
Create an inventory order line.....	32
Create a service tech order line.....	33
Create a task order line.....	34
Create an inspection order line.....	35
<b>Create a customer</b> .....	36
<b>Create an employee (Service Pro user)</b> .....	38
<b>Create a service tech</b> .....	39
<b>Create a master inventory record</b> .....	41
<b>Create an inventory warehouse</b> .....	42



<b>Create a warehouse</b> .....	43
<b>Update records</b> .....	44
<b>Update an appointment</b> .....	45

A large, bold, blue number '1' is centered within a white circle. The circle has a thin blue border and a subtle drop shadow, giving it a three-dimensional appearance. This graphic serves as a section indicator for the first part of the document.

## Introduction to Service Pro API

The Service Pro API is a RESTful client that uses Open Data Protocol (OData) to query results. This guide provides information to help you get started using the API.

The examples in this guide demonstrate how you can use the API to get data from Service Pro, as well as create and update records. In each example, a sample API request and response is shown. Use these examples to help you create API calls and understand the responses that are returned.

A comprehensive list of API controllers, classes, and error codes is available online through the Service Pro Help. The Controllers section describes the API calls that you can use. The Classes section describes the fields on each record. The Error Codes section describes all the errors that the API may return. Use the API reference documentation to work with the Service Pro API.

## Guidelines and requirements

---

To access and use the Service Pro API, you must obtain API credentials by purchasing a Service Pro API Connector license from MSI. As part of your authorization agreement, you must abide by the full terms and conditions in the Service Pro API & Developer Policy. For your convenience, some of the usage terms outlined in the full agreement are summarized here, in addition to other relevant guidelines for general API usage.

 The information provided in this documentation does not describe in full or replace the terms set forth in the official Service Pro API & Developer Policy.

## Supported API call methods

The Service Pro API supports the following API call methods:

- GET – Retrieve data from Service Pro
- POST – Create a record in Service Pro
- PATCH – Partially update or modify data on an existing record in Service Pro
- DELETE – Delete a record from Service Pro

## Service Pro API URL

```
https://servicepro10.com/service/api/
```

The examples in this documentation use relative URLs for the resource you are requesting.

## Rate limits

There is currently no concrete rate limit enforced. However, MSI tracks and compares API counts over time and may enforce limits for companies with unusually high rate consumption based on the number of Service Pro seat licenses associated with the environment.

## CPU time limits

MSI also tracks the amount of time a server's processor is used to process requests (CPU time) for each API key. This measure helps to qualify and quantify the resource demands on the Service Pro API.

There is currently no concrete CPU time limit enforced. However, MSI may enforce limits for companies with unusually high CPU time usage based on the number of Service Pro seat licenses associated with the environment.

## Retries

Your application is permitted to retry failed requests, but unlimited retries are not permitted. The maximum number of retries for an individual data point is 5 times, and the request retry should have an exponential backoff.

## DateTime format requirements

All Service Pro API requests that include datetime strings must use UTC (Zulu) format.

**Example:** 2019-06-10T00:00:00Z

## CorrelationId and CorrelationGroupId headers

MSI asks that all partners use correlation and correlation group id headers when sending requests to the Service Pro API. These headers allow MSI to trace a request from the moment it enters the Service Pro API gateway, through every layer of the application, until the response is returned.

- » A CorrelationId should be a unique string for every request.
- » A CorrelationGroupId should be a shared string for a group of requests.

**Example:** Fetching all records across multiple pages. Each request may only select 10,000 records and each has its own unique CorrelationId, but all share the same CorrelationGroupId.

### *How to send a CorrelationId and CorrelationGroupId*

Both are simply HTTP request headers.

- » The CorrelationId header key is "correlation-id"
- » The CorrelationGroupId header key is "correlation-group-id"

### Curl Example

```
curl --location --request GET
'https://servicepro10.com/service/api/Order?$top=1&$select=Id' \
--header 'correlation-id: 6ea0f9ee-a77c-49cb-97ad-a01808149741' \
--header 'correlation-group-id: 34e36d15-ae36-4c34-ad7f-de11afd447c7' \
--header 'APIKey: example-api-key' \
--header 'SK: example-sk' \
--header 'companyid: example-company-id' \
--header 'accept: application/json, text/javascript, */*; q=0.01' \
--header 'accept-language: en-US,en;q=0.9' \
--header 'accept: application/json, text/javascript, */*; q=0.01'
```



## Authentication

---

To use the API, you must log in using your API Key and Secret Key provided by MSI.

### Example request headers

To create API requests, you must set up authentication in the request headers.

```
Content-Type: application/json or application/xml
APIKey: API Key supplied by MSI
SK: Secret Key supplied by MSI
Company-Version: Version of the company
CompanyId: Id of your company
Correlation-Id: GUID to help track through logs
Disable-Deferreds: boolean (default this to true unless you will be using
deferred to walk the tree)
```

A large blue number '2' is centered within a white circle that has a blue glow effect. This circle is positioned in the upper left area of the page, above the main content box.

## Get data

This chapter explains how to retrieve various information through the API using the GET request method.

*Get order data*

*Get attachments*

*Get Service Pro downtime information*

*About VO classes*

*Select and expand using OData*

## Get order data

The examples in this section show various techniques for retrieving order data through the API.

*Get an order using the order number*

*Get an order using the order ID*

*Include deferred data for an order*

**Note:** Because a GET request retrieves data, unlike a POST or PATCH request, an OData2 query to a specific endpoint may be used in place of a JSON request body.

### Get an order using the order number

#### Example request

To get a specific order number, for example, 32110, use the following URL:

```
/Order
 ?$filter=
  OrderNumber eq '32110'
```

#### Example response – match found

If a match is found, the API responds with a result like this.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2017-08-14-20-58-38",
    "results": [
      {
        "Id": 1,
        "OrderNumber": "32110",
        "CustomerId": 65,
        "ApplyToInvoiceId": null,
        "ShipToAttention": null,
        "ShipToName": "Acacia Supply",
        ...
      }
    ],
    "Message": "Orders Retrieved",
    "Errors": {
      "Validation": [],
      "General": []
    },
    "Warnings": {
```

```
        "General":[],
        "UnknownFields":[]
    },
    "Affirmation":null
}
```

### Example response – match not found

If a match is not found, the API responds with a result like this.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2017-08-14-20-58-38",
    "results": [ ],
    "Message":"Orders Retrieved",
    "Errors":{
      "Validation":[],
      "General":[]
    },
    "Warnings":{
      "General":[],
      "UnknownFields":[]
    },
    "Affirmation":null
  }
}
```

## Get an order using the order ID

### Example request

To get a specific order ID, for example, 1234, use the following URL:

```
/Order/1234
```

### Example response – match found

If a match is found, the API responds with a result like this.

**Note:** These results are not an array.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2017-08-14-20-58-38",
    "results": {
      "Id": 1234,
      "OrderNumber": "1211",
      "CustomerId": 68,
      "ApplyToInvoiceId": null,
      "ShipToAttention": null,
      "ShipToName": "Acme",
      ...
    }
  },
  "Message": "Order Retrieved",
  "Errors": {
    "Validation": [],
    "General": []
  },
  "Warnings": {
    "General": [],
    "UnknownFields": []
  },
  "Affirmation": null
}
```

### Example response – match not found

If a match is not found, the API responds with a 404 error and a message letting you know that the Order Id was not found.

## Include deferred data for an order

If you need to get additional order data beyond what was included in the earlier examples, you can set an additional option in the API request header. When you set the `DisableDeferred` option to `false`, the API response returns all navigation properties for the order. This includes information such as the segments and service items on the order.

After using this method to retrieve the additional order data, you can either `$expand` or make another query to get data for the deferred properties.

### Example request

To get the order segments and service items on an order, use the same URL you used to retrieve the order, but change the request header to include deferred properties:

```
/Order/1234
{
  "Disable-Deferreds": "true"
}
```

### Example response

The new objects that appear in the response contain either a deferred link or a null value. In the example below, the deferred link provides a URL that you can use to get the order segment. The null value informs you that there are currently no service items on the order.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2017-08-14-20-58-38",
    "results": {
      "Id": 1234,
      "OrderNumber": "1211",
      "CustomerId": 68,
      "ApplyToInvoiceId": null,
      "ShipToAttention": null,
      "ShipToName": "Acme",
      "OrderSegment": {__deferred: "OrderSegment?$filter=OrderId eq 1234"},
      "ServiceItem": null
    }
  },
  "Message": "Order Retrieved",
  "Errors": {
    "Validation": [],
    "General": []
  },
  "Warnings": {
    "General": [],
    "UnknownFields": []
  },
  "Affirmation": null
}
```

```
}
```

## Get attachments

This section explains how to download order note attachments and inspection result attachments through the API.

[Download an order note attachment](#)

[Download an inspection result](#)

### Download an order note attachment

You can use the API to download a file that has been uploaded as an order note attachment.

**Note:** This two-step process is similar to downloading an inspection result attachment.

#### Step 1: Get the AttachmentId

Before you can find an order note attachment, you must use a GET request to find the attachment ID based on the order ID. In this example, the OrderId is 228.

#### Example request

```
/Order/RelatedNotes
?$filter=
  OrderId eq 228
```

#### Example response

The AttachmentId is the critical piece of information returned in the API response.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2020-09-08-21-05-45",
    "Results": [
      {
        "Id": 109,
        "OrderId": 228,
        "NoteTypeId": 1,
        "Contents": "493808.png",
        "CreatedGroup": null,
        "LastUpdatedUserId": 1,
        "CreatedUserId": 1,
        "AttachmentId": 45,
        "LastUpdatedDateTime": "2020-09-08T20:58:57Z",
        "CreatedDateTime": "2020-09-08T20:58:57Z",
        "DeletedDateTime": null,
      }
    ]
  }
}
```

```
        "IsDeleted":false,
        "DestinationType":"Order",
        "VistaSMCo":null,
        "MobileCreatedDateTime":null,
        "MobileLastUpdatedDateTime":null
    }
]
},
"Message":"Notes returned",
"Errors":{
    "Validation":[],
    "General":[]
},
"Warnings":{
    "General":[],
    "UnknownFields":[]
},
"Affirmation":null
}
```

### Step 2: Get the attachment

You can use the attachment ID obtained in step 1, along with a GET request, to download the attachment. The response from the API is a byte array of the file.

#### Example request

```
/Attachment/GetAttachment
?id=
45
```

## Download an inspection result

You can use the API to download a file that has been uploaded as an inspection result attachment.

**Note:** This two-step process is similar to downloading an order note attachment. Only step 1 is different.

### Step 1: Get the AttachmentId

Before you can find an inspection result attachment, you must use a GET request to find the attachment ID based on the inspection order line ID.

### Example request

```
/OrderLine/163
?$expand=
  InspectionResult
&$select=
  InspectionResult/AttachmentId
&$top=
  1
```

### Example response

```
{
  "d":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-09-08-21-36-04",
    "Results":{
      "InspectionResult":{
        "AttachmentId":46,
        "Id":15
      },
      "InspectionResultId":15,
      "Id":163
    }
  },
  "Message":"OrderLine Retrieved",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Step 2: Get the attachment

You can use the attachment ID obtained in step 1, along with a GET request, to download the attachment. The response from the API is a byte array of the file.

### Example request

```
/Attachment/GetAttachment  
  ?id=  
    46
```

## Get Service Pro downtime information

Service Pro is routinely updated with new features. When new features are released, MSI sets planned maintenance hours, often with a period of scheduled downtime. During this downtime, requests to the Service Pro API are likely to fail.

Planned maintenance windows can always be viewed at <https://servicepro10.status.io/> at least one week in advance of an upgrade. You can also retrieve this information from the Service Pro API.

### Tips

- » Requests to Service Pro should not be sent during a specified downtime.
- » Integrations into Service Pro should query for potential downtime windows at least once a day.
- » In the very rare occurrence that a downtime has to be extended past the original expiration datetime, it is recommended that all integrations query for a downtime when starting backup post-downtime.

### Example request

To check for planned maintenance, send the following GET request.

```
/Options
 ?$filter=
  OptionNumber eq 'DowntimeWindow'
  &$select=
  Value
```

### Example response – No downtime currently scheduled

If there is no downtime window currently scheduled, the results array is empty.

```
{
  "d":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-11-09-19-38-30",
    "Results":[]
  },
  "Message":"Options Retrieved",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
}
```

```
"Affirmation":null
}
```

### Example response – Downtime currently scheduled

If a maintenance window is currently scheduled, the response looks similar to this.

**Note:** The value field should be split on two pipe characters. The first half is the start datetime (UTC), and the second half is the ending datetime (UTC). If the downtime specified is in the past, it can be ignored.

```
{
  "d":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-11-09-19-38-30",
    "Results":[{
      "Id": 543,
      "Value": "2020-11-09T06:00:00.000Z|2020-11-10T06:00:00.000Z"
    }]
  },
  "Message":"Options Retrieved",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## About VO classes

A comprehensive list of API controllers, classes, and error codes is available online through the Service Pro Help. The Controllers section describes the API calls that you can use. The Classes section describes the fields on each record. The Error Codes section describes all the errors that the API may return. Use the API reference documentation to work with the Service Pro API.

You may notice that many classes have two versions.

<div data-bbox="204 667 571 1088" style="border: 1px solid black; padding: 5px;"> <p><b>Classes</b></p> <ul style="list-style-type: none"> <li>AROpen</li> <li>AROpenVO</li> <li>Accessory</li> <li>AccessoryVO</li> <li style="border: 2px solid orange; padding: 2px;">AccountType</li> <li style="border: 2px solid orange; padding: 2px;">AccountTypeVO</li> <li>AccountsReceivable</li> <li>AccountsReceivableVO</li> </ul> </div>	<p>For each class,</p> <ul style="list-style-type: none"> <li>» The base name describes a Service Pro record and lists all the fields on the record. Use this information to create your API calls.</li> </ul> <div data-bbox="660 719 1362 824" style="border: 1px solid blue; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p><i>Example:</i> The Appointment class describes the fields on the Appointment record.</p> </div> <ul style="list-style-type: none"> <li>» The VO name describes additional related records that you can use to expand your API calls. VO classes provide supplemental information and do not represent actual records in Service Pro. Use this information to expand your API calls and get more information that is not on the base record.</li> </ul> <div data-bbox="660 1077 1362 1223" style="border: 1px solid blue; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p><i>Example:</i> The Appointment VO class describes related records and information that is tied to the Appointment record.</p> </div>
---	--

## Select and expand using OData

---

OData allows you to apply parameters to your GET requests. \$expand allows you to get related entities, while \$select allows you to get a subset of properties.

### Get order segments and order lines

These examples demonstrate how you can get order segments and order lines with an \$expand.

#### Example request

To return only the segment number for the order segment, use this URL:

```
/Order/1234
?$expand=
  OrderSegment
&$select=
  OrderSegment/SegmentNumber
```

#### Example request

To return the expanded order lines from an order segment, use this URL:

```
/Order/1234
?$expand=
  OrderSegment,
  OrderSegment/OrderLine
&$select=
  OrderSegment/OrderLine/ItemNumber
```

#### Example response

The API responds with the expanded order lines for the order segment.

```
{
  "d" : {
    "__count" : null,
    "__next" : null,
    "__prev" : null,
    "__clock" : "2017-12-08-21-00-49",
    "results" : {
      "OrderSegment" : [{
        "SegmentNumber" : "1",
        "OrderLine" : [
          {
            "ItemNumber" : "Oil Change",
            "Id" : 3534
          }
        ],
        "Id" : 254
      }
    ]
  }
}
```

```
        }
      ],
      "Id" : 1234
    }
  },
  "Message" : "Order Retrieved",
  "Errors" : {
    "Validation" : [],
    "General" : []
  },
  "Warnings" : {
    "General" : [],
    "UnknownFields" : []
  },
  "Affirmation" : null
}
```

A large, bold, blue number '3' centered within a white circle. The circle has a thin blue border and a subtle drop shadow, giving it a three-dimensional appearance. It is located in the upper left quadrant of the page.

## Create records

This chapter explains how to create new records through the API using the POST request method.

The examples in this chapter demonstrate various techniques for creating records such as orders, employees, and service techs.

*Create orders*

*Create an order note*

*Create order lines*

*Create a customer*

*Create an employee (Service Pro user)*

*Create a service tech*

*Create a master inventory record*

*Create an inventory warehouse*

*Create a warehouse*



## Create orders

---

The examples in this section show various techniques for creating an order through the API.

*Create an order for a specific customer*

*Create an order for a specific piece of equipment*

*Create an order using a single POST request*

## Create an order for a specific customer

When you create an order from a CustomerId, you send data to the API in the form of a query, which contains additional information at the end of the URL.

### Example request

In this example, you are creating a new order for customer ID 1223 with the order created date that you specify. The system automatically generates the order number when creating the order.

```
https://servicepro10.com/service/api/Order/OrderFromCustomerId/  
1223?orderDate=2018-10-30T00:00:00Z
```

### Example response

Your response, if successful, has a status code of 200 and returns the order you created.

```
{  
  "d": {  
    "__count": null,  
    "__next": null,  
    "__prev": null,  
    "__clock": "2018-10-30-15-32-11",  
    "results": {  
      "Id": 31257,  
      "OrderNumber": "30882",  
      "CustomerId": 1223,  
      ...  
    }  
  },  
  "Message": "Order Created",  
  "Errors": {  
    "Validation": [],  
    "General": []  
  },  
  "Warnings": {  
    "General": [],  
    "UnknownFields": []  
  },  
  "Affirmation": null  
}
```

## Next steps

After creating an order using this method, you can add additional information to the order.

» [Create an order note](#)

» [Create order lines](#)

## Create an order for a specific piece of equipment

When you create an order from a EquipmentId, you send data to the API in the form of a query, which contains additional information at the end of the URL.

### Example request

In this example, you are creating a new order for equipment ID 3023 with the order created date that you specify. The system automatically generates the order number when creating the order.

```
https://servicepro10.com/service/api/Order/OrderFromEquipmentId/3023?orderDate=2018-10-30T00:00:00Z
```

### Example response

Your response, if successful, has a status code of 200 and returns the order you created.

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2018-10-30-15-45-28",
    "results": {
      "Id": 31258,
      "OrderNumber": "30883",
      "CustomerId": 92,
      ...
    }
  },
  "Message": "Order Created",
  "Errors": {
    "Validation": [],
    "General": []
  },
  "Warnings": {
    "General": [],
    "UnknownFields": []
  },
  "Affirmation": null
}
```

## Next steps

After creating an order using this method, you can add additional information to the order.

» [Create an order note](#)

» [Create order lines](#)

## Create an order using a single POST request

As an alternative to creating all parts of an order using multiple requests, this section explains how you can create an order along with all order segments, order lines, and appointments, using a single POST request.



### Notes

- » The Order requires either a CustomerId or SiteId. In this example, a CustomerId is created.
- » The OrderNumber must be unique.
- » OrderLine(s) can be created as a Task, Inventory, ServiceTech, Inspection, or Unknown (anything other than one of the previous). In this example, one of each is created.
- » This example sends the minimum number of fields on each entity. You can send as many other fields you want (fields you do not send will get the default value).

### Example request

```
https://servicepro10.com/service/api/Order
```

### Example request body (required fields)

```
{
  CustomerId: 1,
  OrderNumber: "SLS-6789",
  ExternalNumber: "48934526",
  Appointment: [
    {
      AppointmentName: "Resident Admission"
    },
    {
      AppointmentName: "Resident Room Transfer"
    }
  ],
  OrderSegment: [
    {
      SegmentNumber: "SLS-6789-1",
      SegmentPercent: 100,
      OrderLine: [
        {
          TaskId: 1
        },
        {
          ServiceTechId: 1
        }
      ]
    }
  ]
}
```

```

        SegmentNumber: "SLS-6789-2",
        SegmentPercent: 100,
        OrderLine: [
            {
                InventoryId: 1
            },
            {
                InspectionId: 1
            },
            {}
        ]
    }
]
}

```

### Example response

```

{
  "d":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2019-06-12-20-28-42",
    "results":35289
  },
  "Message":"Order Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}

```

## Create an order note

---

Once an order is created, you may want to add more information to the order, such as order notes. Adding a note to an order requires the same request headers as creating an order; however, a request body is also required. The request body is where you provide the information you want to POST to the record.

### Example request

```
https://servicepro10.com/service/api/OrderNote
```

### Example request body (required fields)

The request body takes an OrderNoteVO.

```
{
  "Contents": "Example Contents",
  "NoteTypeId": 1,
  "OrderId": 31257
}
```

### Example response

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2018-10-30-16-34-50",
    "results": 944
  },
  "Message": "OrderNote Created",
  "Errors": {
    "Validation": [],
    "General": []
  },
  "Warnings": {
    "General": [],
    "UnknownFields": []
  },
  "Affirmation": null
}
```

## Create order lines

---

The examples in this section show various techniques for creating order lines on an order through the API.

*[Create an inventory order line](#)*

*[Create a service tech order line](#)*

*[Create a task order line](#)*

*[Create an inspection order line](#)*

## Create an inventory order line

You can add an inventory order line to an order through the API by creating a POST request.

- » The request body must contain the OrderSegmentId, InventoryId, and the SourceWarehouseId.
- » If you do not indicate an order line quantity, a quantity of 1 is assumed.
- » You can include additional OrderLine fields, such as price or cost. If you do not designate values for these fields, the API uses the values from the inventory record for this order line.

### Example request

```
https://servicepro10.com/service/api/OrderLine
```

### Example request body (required fields)

```
{
  OrderSegmentId: 57204,
  InventoryId: 1,
  SourceWarehouseId: 2
}
```

### Example response

The API responds with a message indicating the order line was successfully created.

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-03-31-16-08-30",
    "Results":93324
  },
  "Message":"OrderLine Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create a service tech order line

You can add a service tech order line to an order through the API by creating a POST request.

- » If you do not indicate an order line quantity, a quantity of 1 is assumed.
- » You can include additional OrderLine fields, such as price or cost. If you do not designate values for these fields, the API uses the values from the service tech record for this order line.

### Example request

```
https://servicepro10.com/service/api/OrderLine
```

### Example request body (required fields)

```
{
  ServiceTechId: 1155,
  OrderSegmentId: 57204,
  Quantity: 1
}
```

### Example response

The API responds with a message indicating the order line was successfully created.

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-03-31-16-12-43",
    "Results":93326
  },
  "Message":"OrderLine Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create a task order line

You can add a task order line to an order through the API by creating a POST request.

- » If you do not indicate an order line quantity, a quantity of 1 is assumed.
- » You can include additional OrderLine fields, such as price or cost. If you do not designate values for these fields, the API uses the values from the task record for this order line.

### Example request

```
https://servicepro10.com/service/api/OrderLine
```

### Example request body (required fields)

```
{
  TaskId: 1,
  OrderSegmentId: 57204,
  Quantity: 1
}
```

### Example response

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-03-31-16-15-50",
    "Results":93328
  },
  "Message":"OrderLine Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create an inspection order line

You can add an inspection order line to an order through the API by creating a POST request.

- » If you do not indicate an order line quantity, a quantity of 1 is assumed.
- » You can include additional OrderLine fields, such as price or cost. If you do not designate values for these fields, the API uses the values from the inspection record for this order line.

### Example request

```
https://servicepro10.com/service/api/OrderLine
```

### Example response

```
{
  InspectionId: 503,
  OrderSegmentId: 57204,
  Quantity: 1
}
```

### Example response

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-03-31-16-19-10",
    "Results":93330
  },
  "Message":"OrderLine Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create a customer

In addition to transactional data such as orders and order lines, you can also use the API to create master records, such as a customer with a default site and ship to.

### Create the customer record

Create a customer record using a POST request.

**Note:** A default site (BillTo) and a default ship to will be created for the customer.

### Example request

```
https://servicepro10.com/service/api/Customer
```

### Example request body (required fields)

```
{
  AccountDate: "2019-06-10T00:00:00Z",
  BillTo: {
    ShipTo: {
      DefaultSalespersonId: 702,
      DefaultWarehouseId: 1,
      ShipToTaxCode: [
        {
          TaxCodeId: 22
        }
      ],
      ShipViaId: 230
    },
    SiteStartDate: "2019-06-10T14:01:44.523Z"
  },
  CreditHold: false,
  CreditLimit: 0,
  CustomerName: "LevySoft",
  CustomerNumber: "95475646456850",
  CustomerSalesperson: [
    {
      SalespersonId: 702
    }
  ],
  CustomerTypeId: 230,
  DiscountPercent: 0,
  DivisionId: 3,
  ExemptNumber: "",
  FinanceCharge: false,
  LinkSortName: false,
  PTD_Costs: 0,
  PTD_Sales: 0,
  SortName: "LevySoft",
}
```

```
StatementFrequencyId: 1,  
Taxable: true,  
TermsId: 22,  
YTD_Costs: 0,  
YTD_Sales: 0  
}
```

## Example response

```
{  
  "d":{  
    "__count":null,  
    "__next":null,  
    "__prev":null,  
    "__clock":"2019-06-10-14-01-55",  
    "results":2266  
  },  
  "Message":"Customer Created",  
  "Errors":{  
    "Validation":[],  
    "General":[]  
  },  
  "Warnings":{  
    "General":[],  
    "UnknownFields":[]  
  },  
  "Affirmation":null  
}
```

## Create an employee (Service Pro user)

You can also use the API to add a new user to the Service Pro company that you are currently logged in to.

- » If this is the first time the user has been added to Service Pro, a new user account is created. All subsequent companies the user is added to use the same account and credentials. The user gets an email with a link to set up a password, and the user cannot log in to Service Pro until after clicking the link and setting up credentials.
- » If the user's account already exists and you are simply adding an existing user to a new company, the user gets an email notification that they have been added to a new company. Clicking a link to establish an account is not necessary. The next time the user logs in, they are asked which company they want to log in to.

### Example request

```
https://servicepro10.com/service/api/CompanyUser/AddUser
```

### Example request body (required fields)

```
{
  "DefaultRegionId": 435,
  "DivisionId": null,
  "EmailAddress": "abc@abc.xyz",
  "EmployeeNumber": "abc.xyz",
  "FirstName": "abc",
  "LastName": "xyz"
}
```

### Example response

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2018-10-30-16-34-50",
    "results": 6705
  },
  "Message": "User Added",
  "Errors": {
    "Validation": [],
    "General": []
  },
  "Warnings": {
    "General": [],
    "UnknownFields": []
  },
  "Affirmation": null
}
```

## Create a service tech

To use Service Pro Mobile, a user must set up as a service tech. You can create a service tech record through the API using a POST request.



### Notes

- » The **EmployeeId** is not a random identifier; rather, it is a foreign key that links the new service tech record to an employee entity. You can create a service tech record without linking to an employee ID; however, the service tech must be associated with an employee in the company in order to log in.
- » The **Schedule** field is a toggle that determines whether the service tech appears on the Scheduler board.

### Example request

```
https://servicepro10.com/service/api/ServiceTech
```

### Example request body (required fields)

```
{
  "EmailAddress": "abc@abc.xyz",
  "EmployeeId": 41887,
  "OnStaff": true,
  "OvertimeCost": 0,
  "OvertimeFactor": 0,
  "OvertimeRate": 0,
  "SACId": 379,
  "Schedule": true,
  "ServiceTechName": "abc xyz",
  "ServiceTechNumber": "abc.xyz",
  "SourceWarehouseId": 2,
  "SpecialCost": 0,
  "SpecialFactor": 0,
  "SpecialRate": 0,
  "StandardCost": 0,
  "StandardRate": 0
}
```

### Example response

```
{
  "d": {
    "__count": null,
    "__next": null,
    "__prev": null,
    "__clock": "2018-10-30-16-34-50",
    "results": 1487
  },
}
```

```
"Message": "ServiceTech Created",
"Errors": {
  "Validation": [],
  "General": []
},
"Warnings": {
  "General": [],
  "UnknownFields": []
},
"Affirmation": null
}
```

## Create a master inventory record

---

You can create an inventory record through the API by creating a POST request.

### Example request

```
https://servicepro10.com/service/api/Inventory
```

### Example request body (required fields)

```
{
  InventoryNumber: "I-9237",
  Description: "DESC: Inventory",
  SortDescription: "DESC: Inventory",
  DefaultWarehouseId: 2,
  ActivityId: 1,
  ActivityDate: "2020-07-29 00:00:00",
  Serialized: false,
  Lotted: false,
  Taxable: true,
  PayCommission: true,
  UpdateNewEquipment: true,
  Obsolete: false,
  CommissionPercent: 10,
  ShelfLifeDays: 0,
  LinkSortDescription: true
}
```

### Example response

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-07-30-00-42-47",
    "Results":7
  },
  "Message":"Inventory Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create an inventory warehouse

---

You can create an inventory warehouse through the API by creating a POST request.

### Example request

```
https://servicepro10.com/service/api/InventoryWarehouse
```

### Example request body (required fields)

```
{
  InventoryId: 7,
  WarehouseId: 1,
  AssetCostTypeId: 1,
  SACId: 1
}
```

### Example response

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-07-30-00-50-05",
    "Results":1
  },
  "Message":"InventoryWarehouse Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

## Create a warehouse

---

You can create a warehouse through the API by creating a POST request.

### Example request

```
https://servicepro10.com/service/api/Warehouse
```

### Example request body (required fields)

```
{
  WarehouseNumber: "I0100",
  WarehouseIsVan: false
}
```

### Example response

```
{
  "D":{
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-07-30-00-18-22",
    "Results":2
  },
  "Message":"Warehouse Created",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```

A large blue number '4' centered within a white circle that has a blue border. This circle is part of a larger blue-bordered rounded rectangle that frames the main content area of the page.

## Update records

This chapter explains how to update existing records through the API using the PATCH request method.

To successfully update an existing record using a PATCH request, the request body must include the Id of the entity that you want to update and **ONLY** the data that you are updating.

### *Update an appointment*

## Update an appointment

---

In this example, an appointment has been created, and you want to update the appointment name.

### Example request

```
https://servicepro10.com/service/api/Appointment
```

### Example request body (required fields)

The request body must identify the ID number of the appointment you want to update. In this example, you are updating the AppointmentName for appointment 335.

```
{
  Id: 335,
  AppointmentName: "TA-387"
}
```

### Example response

The API responds with a message indicating the appointment was successfully updated.

```
{
  "D": {
    "__count":null,
    "__next":null,
    "__prev":null,
    "__clock":"2020-03-04-15-05-08",
    "Results":335
  },
  "Message":"Appointment Updated",
  "Errors":{
    "Validation":[],
    "General":[]
  },
  "Warnings":{
    "General":[],
    "UnknownFields":[]
  },
  "Affirmation":null
}
```