

# TM Forum Specification

## Service Ordering Management API Conformance Profile

**TMF641B**  
**Release 18.0.1**  
**May 2018**

<b>Latest Update: TM Forum Release 18.0.1</b>	<b>TM Forum Approved</b>
<b>Version: 1.0.1</b>	<b>IPR Mode: RAND</b>

## NOTICE

Copyright © TM Forum 2018. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to TM FORUM, except as needed for the purpose of developing any document or deliverable produced by a TM FORUM Collaboration Project Team (in which case the rules applicable to copyrights, as set forth in the [TM FORUM IPR Policy](#), must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by TM FORUM or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and TM FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

TM FORUM invites any TM FORUM Member or any other party that believes it has patent claims that would necessarily be infringed by implementations of this TM Forum Standards Final Deliverable, to notify the TM FORUM Team Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this deliverable.

The TM FORUM invites any party to contact the TM FORUM Team Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this TM FORUM Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this TM FORUM Standards Final Deliverable. TM FORUM may include such claims on its website but disclaims any obligation to do so.

TM FORUM takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this TM FORUM Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on TM FORUM's procedures with respect to rights in any document or deliverable produced by a TM FORUM Collaboration Project Team can be found on the TM FORUM website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this TM FORUM Standards Final Deliverable, can be obtained from the TM FORUM Team Administrator. TM FORUM makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100

Parsippany, NJ 07054, USA

Tel No. +1 973 944 5100

Fax No. +1 973 944 5110

TM Forum Web Page: [www.tmforum.org](http://www.tmforum.org)

## TABLE OF CONTENTS

NOTICE .....	2
TABLE OF CONTENTS .....	4
LIST OF TABLES.....	5
INTRODUCTION.....	6
API DESCRIPTION .....	7
RESOURCE MODEL CONFORMANCE.....	8
ServiceOrdering API MANDATORY AND OPTIONAL RESOURCES.....	8
ServiceOrder resource MANDATORY AND OPTIONAL ATTRIBUTES.....	8
API OPERATIONS CONFORMANCE.....	12
SERVICE ORDER MANDATORY AND OPTIONAL OPERATIONS.....	12
API GET FILTERING OPERATION CONFORMANCE.....	13
Filtering in ServiceOrder resource .....	13
GET /ServiceOrderingManagement/ServiceOrder/ .....	15
GET /ServiceOrderingManagement /ServiceOrder/{ID} .....	15
API POST OPERATION CONFORMANCE .....	16
POST /ServiceOrderingManagement /ServiceOrder/ .....	16
API PATCH OPERATION CONFORMANCE.....	19
API DELETE OPERATION CONFORMANCE.....	20
API CONFORMANCE TEST SCENARIOS.....	21
ServiceOrder resource TEST CASES.....	21
ACKNOWLEDGEMENTS.....	30
Release History .....	30

## LIST OF TABLES

N/A

## INTRODUCTION

The following document is the REST API Conformance for the Service Ordering API (TMF 641).

## API DESCRIPTION

A service order will describe a list of service order items. A service order item references an action on an existing or future service. By service we designed Customer-Facing Service (CFS) as well as Resource Facing Service (RFS).

A service order is created based on service(s) that is(are) defined in a catalog.

**RESOURCE MODEL CONFORMANCE**

## ServiceOrdering API MANDATORY AND OPTIONAL RESOURCES

Resource Name	Mandatory or Optional	Comments
ServiceOrder	M	

## ServiceOrder resource MANDATORY AND OPTIONAL ATTRIBUTES

Attribute Name	Mandatory or Optional	Comments
id	M (in response messages) O (otherwise)	Initially generated by the server and provided in the response upon resource creation.  Not available in POST request
href	M (in response messages) O (otherwise)	Initially generated by the server and provided in the response upon resource creation  Value in response must be the same as the one set in Location header provided upon entity creation  Not available in POST request
externalId	O	
priority	O	Implementation could defined a be default value
description	O	
category	O	
state	M (in response messages) O (otherwise)	Only required in response messages from the server when providing the ServiceOrdering entity contents (responses to create-POST and read-GET the entity)



Attribute Name	Mandatory or Optional	Comments
orderDate	M (in response messages)	Not available in POST request
completionDate	O	Not available in POST request
requestedStartDate	O	
requestedCompletionDate	O	
expectedCompletionDate	O	Not available in POST request
startDate	M (in response messages)	Not available in POST request
NotificationContact	O	
@baseType	O	
@type	O	
@schemaLocation	O	
relatedParty	O	Array of structures
id	M except if href provided	
href	M except if id provided	
role	M	
name	O	
@referredType	O	
orderRelationship	O	Array of structures
type	M	
id	M except if href provided	
href	M except if id provided	
@referredType	O	
note	O	Array of note(s)
date	O	
author	M	
text	M	
serviceOrderItem	M	Array of item – At least one must be provided
id	M	
action	M	
state	M (in response messages) O (otherwise)	Only required in response messages from the server when providing the ServiceOrdering entity contents (responses to create-POST and read-GET the entity)
@type	O	

Attribute Name	Mandatory or Optional	Comments
@schemaLocation	O	
@baseType	O	
appointment	O	Array of appointment(s)
id	M except if href provided	
href	M except if id provided	
@referredType	O	
orderItemRelationship	O	Array of Item relationship
type	M	
id	M	
Service		0 or 1
id	M except if href provided	
href	M except if id provided	
name	O	
serviceState	O	
@type	O	
@schemaLocation	O	
ServiceSpecificationRef	O	0 or 1
id	M	
href	M	
version	O	
name	O	
@type	O	
@schemaLocation	O	
@baseType	O	
TargetServiceSchema		0 or 1
@type	M	
@schemaLocation	M	
Place	O	Array of place(s)
id	M except if href provided	
href	M except if id provided	
name	O	
role	M	
@referredType	M	Because place could be a geographicAddress, geographicLocation or a geographicSite, a type must be provided.

Attribute Name	Mandatory or Optional	Comments
@schemaLocation	O	
ServiceCharacteristic		Array of product characteristic(s)
name	M	
valueType	M	
Value	M	1 – one value must be provided for each serviceCharacteristic
@type	M	
@schemaLocation	M	
ServiceRelationship		Array of service relationship(s)
type	M	
(service) targeted	M	A referred or valued service must be present

**API OPERATIONS CONFORMANCE**

For every single resource use the following templates and define what operations are optional and what operations are mandatory.

**SERVICE ORDER MANDATORY AND OPTIONAL OPERATIONS**

Uniform API Operation	Mandatory/Optional	Comments
GET	M	GET must be used to retrieve a representation of a resource
POST	M	POST must be used to create a new resource
PUT	O	PUT must be used to completely update a resource identified by its resource URI
PATCH (JSON-MERGE)	O	PATCH must be used to partially update a resource
DELETE	O	DELETE must be used to remove a resource

## API GET FILTERING OPERATION CONFORMANCE

### Definitions

**Filtered Search:** A filtered search can be applied using query parameters in order to obtain only the resource entities that meet the criteria defined by the filtering parameters included in the query request. Several elements can be applied to the filtered search. In that case logic, a logical AND is applied to combine the criteria (e.g.:?severity=<value> &status=<value>)

**Filtered Data (Attribute selection):** In order to apply a filter and limit the number of attributes included in the response, the GET request can include the “?fields=” query parameter. Several elements can be applied to the filter. In that case, a logical AND is applied to combine the values (e.g.:?fields=category,state) will provide in the response only the values assigned to attributes category and state. Attribute selection capabilities are the same for collections retrieval and individual resource queries

### Filtering in ServiceOrder resource

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
id	M	M	M	M
href	NA	NA	O	O
externalId	M	O	M	O
priority	O	O	O	O
description	O	NA	M	NA
category	O	NA	O	NA
state	M	NA	M	NA
orderDate	M	NA	O	O
completionDate	M	NA	O	O
requestedStartDate	O	NA	M	NA
requestedCompletionDate	O	NA	O	NA
expectedCompletionDate	O	NA	M	NA
startDate	O	NA	M	NA
notificationContact	O	NA	O	NA
<b>OrderRelationship</b>				
id	O	NA	O	O
type	O	NA	O	NA
href	NA	NA	O	NA

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
<b>RelatedParty</b>				
id	O	NA	O	O
href	NA	NA	NA	NA
role	O	NA	O	O
name	O	NA	O	NA
validFor	NA	NA	NA	NA
<b>ServiceOrderItem</b>				
id	NA	NA	M	O
state	O	NA	M	O
action	O	NA	O	O
<b>OrderItemRelationship</b>				
id	NA	NA	NA	NA
type	NA	NA	NA	NA
<b>serviceSpecificationRef</b>				
id	O	NA	O	NA
href	NA	NA	NA	NA
version	O	NA	O	NA
name	O	NA	O	NA
<b>service</b>				
id	O	NA	O	NA
href	NA	NA	NA	NA
name	O	NA	O	NA
serviceState	O	NA	O	NA
<b>place</b>				
id	O	NA	O	NA
href	NA	NA	NA	NA
role	O	NA	O	NA
name	O	NA	O	NA
<b>appointment</b>				
id	O	NA	O	O
href	NA	NA	O	NA
<b>note</b>				

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
date	NA	NA	NA	NA
author	NA	NA	NA	NA
text	NA	NA	NA	NA

## GET /ServiceOrderingManagement/ServiceOrder/

**Filtered Search:** A filtered search can be applied using the following filtering criteria

Providing filtering criteria is mandatory to avoid too many answers retrieved. See example below:

- externalId
- description
- orderDate
- state + orderDate interval (orderDate > 11/4/2018 & orderDate > 11/9/2017)

**Filtered Data:** A filtered response can be requested for the following attributes using the “?fields=” query parameter

- Any of the attributes in the first level of ServiceOrdering resource definition

## GET /ServiceOrderingManagement /ServiceOrder/{ID}

**Filtered Search:** A filtered response can be requested for the following attributes using the “?fields=” query parameter

- Any of the attributes in the first level of ServiceOrder resource definition

## API POST OPERATION CONFORMANCE

### POST /ServiceOrderingManagement /ServiceOrder/

This Uniform Contract operation is used to create a ServiceOrder resource in the server.

The response to this operation must include a Location header set to /ServiceOrderingManagement /ServiceOrder/{ID} where {ID} indicates the identifier assigned by the server to the new ServiceOrder resource created

POST	M	
Response Status Code 201	M	
Other Status Codes	NA	

The following table indicates attributes that are required to be sent when creating a new ServiceOrder as well as attributes with special considerations. All other attributes defining the resource are not required to be sent as part of the BODY of the POST request message:

Attribute name	Mandatory	Default	Rule
id	NA		Must not be in POST request
href	NA		Must not be in POST request
externalId	O		
priority	O	"4"	Could be implementation-based defined
description	O		
category	O		
state	NA		Must not be in POST request
orderDate	NA		Must not be in POST request
completionDate	NA		Must not be in POST request
requestedStartDate	O		
requestedCompletionDate	O		
expectedCompletionDate	NA		Must not be in POST request



Attribute name	Mandatory	Default	Rule
startDate	NA		
NotificationContact	O		
note	O		If note present, author and text are mandatory. Date is optional and values by default with today date.
orderRelationship	O		If orderRelationship present, type and (id AND/OR href) are mandatory
ServiceOrderItem	M		At least one item must be provided.
orderItem.id	M		
orderItem.action	M		
orderItem.state	NA		Must not be in POST request
orderItemRelationship	O		if provided, type & id are mandatory
appointment	O		href or/and id must be provided
relatedParty	M		At least one relatedParty must be provided.  href or/and id must be provided  role is mandatory
place	O		href or/and id must be provided  Role is mandatory
serviceRelationship	O		type is mandatory
serviceSpecification	O		href or/and id must be provided
targetServiceSchema	O		@type & @schemaLocation must be provided
serviceCharacteristic	O		name and valueType must be provided
value	O		@type & @schemaLocation must be provided

The response from the server must include a BODY with the contents of the new resource created, filled with at least the same information elements that were included in the request

and are supported by the server. Notice that the value stored by the server may be different than the one set in the request (e.g.: `serviceSpecification.Name`)

If the POST request includes optional parameters (as per the model resource definition) that are not supported by the server, then the server must reject the request (replying with a 4xx error response) indicating the parameter not supported.

The BODY of the response from the server must include attribute “href” set to the same value as the one in the Location header.

The server must include in the BODY of the response, even if they are not included in the request, the following attributes that are mandatory in the definition of a ServiceOrder as per the resource model defined

- id
- href
- state
- orderDate
- orderItem.state

We must note that some attributes must not be send in the POST request and could be not valued in the POST answer:

- expectedCompletionDate
- completionDate
- startDate

The BODY of the response from the server could include attribute “href” (or “reference”) under each one of the entities within the ServiceOrder model that can be addressed individually and were included in the response. This applies to any of the following entities

- relatedParty
- serviceSpecification
- channel
- place
- service

## API PATCH OPERATION CONFORMANCE

This section defines which attributes are patchable.

Since PATCH operation is optional and not included in the basic CONNECT certification this is not applicable in this conformance document.

## API DELETE OPERATION CONFORMANCE

This section defines what operations can be used to delete a ServiceOrder resource.

Since DELETE operation is optional and not included in the basic CONNECT certification this is not applicable in this conformance document

## API CONFORMANCE TEST SCENARIOS

This section describes the test scenarios required for the basic CONNECT certification of ServiceOrdering API.

Test Cases must be executed in the order defined for each resource because the result from one of the scenarios will be input for the next one.

Requests must be addressed to the endpoint provided for certification, specifically they must be addressed to the URI defined by the concatenation of the {apiRoot} and the specific resource, where the {apiRoot} is defined as {serverRoot}/ServiceOrderingManagement/v1, being {serverRoot} defines the certification endpoint

### ServiceOrder resource TEST CASES

#### Nominal Scenarios

##### TC\_ServiceOrder\_N1 – Create new ServiceOrder with minimum required information

- Send a POST message to {apiRoot}/ServiceOrder/ with the following contents in the BODY

```
{
  "externalId": "OrangeBSS748",
  "priority": "1",
  "description": "Service order description",
  "category": "CloudServiceOrdering",
  "requestedStartDate": "2018-01-15T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
  "@type": "standard",
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "@type": "standard",
      "service": {
        "serviceState": "Active",
        "serviceCharacteristic": [
          {
            "name": "vCPE_IP",
            "valueType": "String",
            "value": {
              "@type": "IPAddress",
              "@schemaLocation": "http://...",
              "vCPE_IP": "193.218.236.21"
            }
          }
        ]
      }
    }
  ],
  "serviceSpecification": {
    "id": "12",
    "href": "http://...:serviceSpecification/12",
```

```
        "name": "vCPE",
        "version": "1",
        "@type": "vCPE",
        "@schemaLocation": "http..."
    }
}
]
}
```

- Wait for a response from the server with the following characteristics
  - Response Code 201-Created
  - Include a location header in the body set to `/{apiRoot}/ServiceOrder/{IDSO1}` where `{IDSO1}` indicates the identifier assigned by the server to the new `ServiceOrder` resource
  - The response message includes all mandatory parameters (including `id`, `state` and `orderDate` that were not sent in the original request)
  - The body of the response matches the values set in the original request
- Send a GET message to `/{apiRoot}/ServiceOrder/{IDSO1}`
- Wait for a response from the server with the following characteristics
  - Response Code 200-OK
  - The body of the response includes one `ServiceOrder` resource with ID set to `{IDSO1}`, the same identifier as assigned by the server to the new resource created
  - The response message includes all mandatory parameters
  - The body of the response for the resource with identifier `{IDSO1}` matches the values set in the original request

**TC\_ServiceOrder\_N2 – Create new ServiceOrder with minimum set of parameters supported by server**

- Send a POST message to {apiRoot}/ServiceOrder/ with the following contents in the BODY

```
{
  "externalId": "OrangeBSS954",
  "priority": "2",
  "description": "Service order description",
  "category": "CloudServiceOrdering",
  "requestedStartDate": "2018-01-18T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-19T09:37:40.508Z",
  "@type": "standard",
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "@type": "standard",
      "service": {
        "serviceState": "Active",
        "serviceCharacteristic": [
          {
            "name": "vCPE_IP",
            "valueType": "String",
            "value": {
              "@type": "IPAddress",
              "@schemaLocation": "http...",
              "vCPE_IP": "193.218.459.78"
            }
          }
        ]
      },
      "serviceSpecification": {
        "id": "12",
        "href": "http://...:serviceSpecification/12",
        "name": "vCPE",
        "version": "1",
        "@type": "vCPE",
        "@schemaLocation": "http..."
      }
    }
  ]
}
```

- Wait for a response from the server with the following characteristics
  - Response Code 201-Created

- Include a location header in the body set to `/{apiRoot}/ServiceOrder/{IDSO2}` where `{IDSO2}` indicates the identifier assigned by the server to the new `ServiceOrder` resource
- The response message includes all mandatory parameters (including `orderDate` and `state` that were not sent in the original request)
- The body of the response matches the values set in the original request
- Send a GET message to `/{apiRoot}/ServiceOrder/{IDSO2}`
- Wait for a response from the server with the following characteristics
  - Response Code 200-OK
  - The body of the response includes one `ServiceOrder` resource with ID set to `{IDSO2}`, the same identifier as assigned by the server to the new resource created
  - The response message includes all mandatory parameters
  - The body of the response for the resource with identifier `{IDSO2}` matches the values set in the original request

### **TC\_ServiceOrder\_N3 – Search for ServiceOrder with specific characteristics**

- Send a GET message to `/{apiRoot}/ServiceOrder?category=CloudServiceOrdering&orderItem.service.serviceSpecification=12`
- Wait for a response from the server with the following characteristics
  - Response Code 200-OK
  - The body of the response includes at least two `ServiceOrder` resources referring to `{IDSO1}` and `{IDSO2}`
  - The body of the response for the resource with each identifier matches the values in the corresponding original request
- Send a GET message to `/{apiRoot}/ServiceOrder?priority=1&category=CloudServiceOrdering`
- Wait for a response from the server with the following characteristics



- Response Code 200-OK
  - The body of the response includes one ServiceOrder resource referring to {IDSO1} and there is no reference to ServiceOrder resource {IDSO2}
  - The response message includes all mandatory parameters
  - The body of the response for the resource with identifier {IDSO1} matches the values in the original request
- Send a GET message to `/{apiRoot}/ServiceOrder?externalId= OrangeBSS954`
  - Wait for a response from the server with the following characteristics
    - Response Code 200-OK
    - The body of the response includes one ServiceOrder resource referring to {IDSO2} and there is no reference to ServiceOrder resource {IDSO1}
    - The response message includes all mandatory parameters
    - The body of the response for the resource with identifier {IDSO2} matches the values in the original request

#### **TC\_ServiceOrder\_N4 – Filtered retrieval of ServiceOrder**

- Send a GET message to `/{apiRoot}/ServiceOrder/{IDSO2}?fields=id,href,externalId,priority,state`
- Wait for a response from the server with the following characteristics
  - Response Code 200-OK
  - The body of the response includes one ServiceOrder resource referring to {IDSO2} and including only attributes id,href,externalId, priority and state.
- Send a GET message to `/{apiRoot}/ServiceOrder/{IDSO1}?fields= id, state, orderItem.id,orderItem.state,orderItem.action`
- Wait for a response from the server with the following characteristics

- Response Code 200-OK
- The body of the response includes one ServiceOrder resource referring to {IDSO1} and including only attributes id, state, and an array of order item with attributes: id, state, and action matching the values in the original request.

Notice that this test case is using parameters "state", "id", "orderItem.id", "orderItem.state" and "orderItem.action" to filter the data included in the response but any other parameter could be used

### **TC\_ServiceOrder\_N5 – Filtered Search and Filtered data response**

- Send a GET message to `/{apiRoot}/ServiceOrder?externalId=OrangeBSS748&fields=id,state,category,description`
- Wait for a response from the server with the following characteristics
  - Response Code 200-OK
  - The body of the response includes one ServiceOrder resource referring to {IDSO1} and there is no reference to ServiceOrder resource {IDSO2}
  - The body of the response for the resource includes only attributes id, state and category matching the values in the corresponding original request

Notice that this test case is using the parameters id,state,category, and description to filter the data included in the response but any other parameter could be used

## **Error Scenarios**

### **TC\_ServiceOrder\_E1 – Unknown Trouble ServiceOrder identifier**

- Send a GET message to `/{apiRoot}/ServiceOrder/{IDSO3}`, where {IDSO3} does not match any of the identifiers previously created in the server
- Wait for a response from the server with the following characteristics
  - Response Code 404-Not Found

### **TC\_ServiceOrder\_E2 – Invalid Request – unexpected parameter**

- Send a POST message to {apiRoot}/ServiceOrder/ with the following contents in the BODY.

```
{
  "externalId": "OrangeBSS777",
  "priority": "1",
  "description": "Service order description",
  "category": "TMF resource illustration",
  "state": "acknowledged",
  "requestedStartDate": "2018-01-15T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
  "expectedCompletionDate": "2018-01-15T09:37:40.508Z",
  "@type": "standard",
  "note": {
    "date": "2018-01-15T09:37:40.508Z",
    "author": "Jean Pontus",
    "text": "bla bla bla"
  },
  "relatedParty": [
    {
      "id": "456",
      "href": "http://serverlocation:port/partyManagement/party/456",
      "role": "requester"
    }
  ],
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "state": "acknowledged",
      "@type": "standard",
      "service": {
        "serviceState": "Active",
        "serviceCharacteristic": [
          {
            "name": "vCPE_IP",
            "valueType": "String",
            "value": {
              "@type": "IPAddress",
              "@schemaLocation": "http....",
              "vCPE_IP": "193.218.236.21"
            }
          }
        ]
      }
    }
  ],
  "serviceSpecification": {
    "id": "12",
    "href": "http://...:serviceSpecification/12",
    "name": "vCPE",
    "version": "1",
    "@type": "vCPE",
    "@schemaLocation": "http..."
  }
}
```

```

    }
  ]
}

```

Notice that this request features unexcepting parameters as state, orderItem.state and expectedCompletionDate.

- Wait for an error response from the server indicating the unexpected parameters in the request

### TC\_ServiceOrder\_E3 – Invalid Request – Missing parameter mandatory in context

- Send a POST message to {apiRoot}/ServiceOrder/ with the following contents in the BODY.

```

{
  "externalId": "OrangeBSS566",
  "priority": "2",
  "description": "Service order description",
  "category": "CloudServiceOrdering",
  "requestedStartDate": "2018-01-18T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-19T09:37:40.508Z",
  "@type": "standard",
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "@type": "standard",
      "service": {
        "serviceState": "Active",
        "serviceCharacteristic": [
          {
            "name": "vCPE_IP",
            "valueType": "String",
            "value": {
              "@type": "IPAddress",
              "@schemaLocation": "http...",
              "vCPE_IP": "193.218.459.78"
            }
          }
        ]
      },
      "serviceSpecification": {
        "name": "vCPE",
        "version": "1",
        "@type": "vCPE",
        "@schemaLocation": "http..."
      }
    }
  ]
}

```

- Notice that this request is missing a service specification id or href in the order item described (action is added so it must be specified) but any other mandatory parameter could be used
  
- Wait for an error response from the server indicating the mandatory parameter is missing in the request

**ACKNOWLEDGEMENTS****RELEASE HISTORY**

Release Number	Date	Release led by:	Description
Release 18.0.0 Version 1.0.0	12-Jan-2018	Ludovic Robert	First Release of the Document.
Release 18.0.1 Version 1.0.1	02-May-2018	Adrienne Walcott	Updated to reflect TM Forum Approved status. Updated to TM Forum new brand guidelines