

January 2014

VoDKA.TV Integration API

OSS

Contents

1. Introduction	3
2. Administration API	3
2.1. Households	6
2.1.1. Create a New Household	6
2.1.2. Find All Households	7
2.1.3. Update a Household	9
2.1.4. Delete a Household	10
2.2. Devices	11
2.2.1. Create a Device	12
2.2.2. Find a Device	14
2.2.3. Find Devices	15
2.2.4. Find the Devices of a Household	17
2.2.5. Find the Devices of a Tag Node	18
2.2.6. Update a Device	20
2.2.7. Delete a Device	22

1. Introduction

This document describes the VoDKA.TV API for integrating with external applications. Thus, it can be used to avoid using the administration interface that VoDKA.TV provides, and to manage the configuration of VoDKA.TV with an external application.

2. Administration API

The VoDKA.TV administration API consists of a set of GET or POST HTTP requests with the appropriate parameters. By default, it is not needed authentication for accessing the URLs of the API. However, the protocol `https` may be used in the communication.

After processing a request, VoDKA.TV always returns an XML response to the external system which made the request. In this response, VoDKA.TV indicates to the external system either the result of the invoked operation or an occurred error. Thus, when an error occurs, the XML response includes an element `errors` which contains the following elements:

- A `code` element which identifies the error. Each API method defines a specific set of error codes.
- A `description` element explaining the cause of the error.
- A set of `param` tags to specify extra data associated to the error.

```
<?xml version="1.0" encoding="UTF-8"?>
...
<errors>
  <error>
    <code>ERROR_CODE</code>
    <params>
      <param name="PARAM_NAME" value="PARAM_VALUE" />
      <param name="PARAM_NAME" value="PARAM_VALUE" />
      ...
    </params>
    <description>ERROR_DESCRIPTION</description>
  </error>
  ...
</errors>
...
```

For example, if an external system tries to create a new room with an already existing ID, the XML response could be:

```
<?xml version="1.0" encoding="UTF-8"?>
...
<errors>
  <error>
    <code>uplicated</code>
    <params>
      <param name="roomId" value="ROOM-001" />
    </params>
  </error>
</errors>
```

```
<description>
  The room ROOM-001 already exists
</description>
</error>
...
</errors>
...
```

There are three special error codes that can be returned by any API method:

Error code	Parameters	Error description
invalid_xml		The supplied XML is not valid (empty or syntax errors).
internal_error		An internal error has occurred while executing the method. Details can be found in the VoDKA.TV logs.
required	<param_id> with the name of the missing parameter.	Some required parameter is not provided in the request.

In addition, all requests can receive an optional parameter that will be present in the XML response. There are two ways to provide that parameter:

- For POST requests, the parameter is a new attribute in the XML sent as a POST parameter. This attribute is called `external`. For instance, if we send a XML like this:

```
<request>
  <param1>
  ...
  <paramN>
  <external>
    <id>861234567</id>
  </external>
</request>
```

we get a response like this:

```
<response>
  <response1>
  ...
  <response>
  <external>
    <id>861234567</id>
  </external>
</response>
```

- For GET requests, the parameter is a new GET parameter called `external`. The value for this parameter will be included in the XML response. Take into account that the value of the GET parameter `external` will be directly included in the XML responses. Therefore, apart from being escaped properly like any other GET parameters, this value must be a valid XML fragment, otherwise the responses will not have a valid XML format, which will

cause problems to XML parsers. For instance, if we invoke a URL like this: `https://vodkatv/param1=xxx&external=%3Cid%3E861234567%3C%2Fid%3E`, we get a response like this:

```
<response>
  <response1>
    ...
  <response>
    <external>
      <id>861234567</id>
    </external>
  </request>
```

In the following sections, this API will be described in detail. For that, we assume that the VoDKA.TV application can be accessed through the URL `https://vodkatv/`.

To describe the API, the specific XML formats used by each call will be explained. When an specific attribute can be empty or optional, it means it is not needed to include this attribute in the XML data. Note that it is completely different to specify an attribute with an empty value from not specifying the entire attribute skipping the corresponding element of the XML file. In the first case the attribute will be assigned to an empty string; in the second case the attribute will be a *null* value.

Optional elements will be marked between square brackets [OPTIONAL ELEMENT]. The rest of the elements in XML formats are assumed to be mandatory. On the other side, for GET parameters all elements will be mandatory by default.

All date parameters used in the API follow the pattern (yyyy-MM-dd'T'HH:mm:ssZ), where:

- All '-' and ':' are literals to be included as-is.
- yyyy: four-digit year.
- MM: two-digit month (range 01-12).
- dd: two-digit day (range 01-31).
- 'T': literal letter 'T', used as separator.
- HH: two-digit hour (range 00-23).
- mm: two-digit minute (range 00-59).
- ss: two-digit second (range 00-59).
- Z: four-digit hour difference with UTC, including '+' or '-' sign (examples -0100, +0200).

A sample would be 2010-05-20T11:14:22+0100.

For those attributes that need to be internationalized is used the *i18nValues* type as in the following example:

```
<i18nValues>
  <i18nValue lang="en">
```

```
        ENGLISH TRANSLATION
    </i18nValue>
    <i18nValue lang="ar">
        ARABIC TRANSLATION
    </i18nValue>
    ...
</i18nValues>
```

The language code should be specified using the appropriate subtags of the IANA Language Subtag Registry

2.1. Households

In the VoDKA.TV system, households are used to group devices (see section 2.2.) in logical units. VoDKA.TV also names *households* as *rooms*, and therefore all the references to *rooms* refer to *households* as well.

A household has the following attributes:

- `roomId` (TEXT(50)): the household identifier.
- `description` (TEXT): the household description (optional).
- `tag`: the node of the tag tree to which this household is associated (see ??) (optional).
 - `tagId` (INTEGER): the identifier of the node of the tag tree.

If present, all the devices of that household, except those devices that redefine this value (see 2.2.), will be associated to this node of the tags tree.

2.1.1. Create a New Household

It creates a new household with the specified values.

URL <https://vodkatv/external/admin/configuration/CreateRoom.do>

Parameters POST parameter with the information of the new household in the following XML format:

```
<?xml version="1.0" encoding="UTF-8"?>
<room>
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    ROOM_DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
```

```
        </tagId>
    </tag>]
</room>
```

Response XML with the new entered household information:

```
<?xml version="1.0" encoding="UTF-8"?>
<room>
    <roomId>
        ROOM_ID
    </roomId>
    [<description>
        ROOM_DESCRIPTION
    </description>]
    [<tag>
        <tagId>
            TAG_TREE_NODE_ID
        </tagId>
        [<name>
            TAG_TREE_NODE_NAME
        </name>]
        [<parentId>
            TAG_TREE_NODE_PARENT_ID
        </parentId>]
        [<position>
            TAG_TREE_NODE_POSITION
        </position>]
    </tag>]
    <!-- This element is included only if an error occurs -->
    <errors>
        ...
    </errors>
</room>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
duplicated	<ul style="list-style-type: none">roomId	A household with the same identifier roomId already exists on the system.

2.1.2. Find All Households

It returns a list of all households defined on the system.

URL <https://vodkatv/external/admin/configuration/FindAllRooms.do>

Parameters None.

Response XML specifying all defined households:

```
<?xml version="1.0" encoding="UTF-8"?>
<rooms>
  <room>
    <roomId>
      ROOM_ID_1
    </roomId>
    [<description>
      ROOM_DESCRIPTION_1
    </description>]
    [<tag>
      <tagId>
        TAG_TREE_NODE_ID
      </tagId>
      [<name>
        TAG_TREE_NODE_NAME
      </name>]
      [<parentId>
        TAG_TREE_NODE_PARENT_ID
      </parentId>]
      [<position>
        TAG_TREE_NODE_POSITION
      </position>]
    </tag>]
  </room>
  <room>
    <roomId>
      ROOM_ID_2
    </roomId>
    [<description>
      ROOM_DESCRIPTION_2
    </description>]
    [<tag>
      <tagId>
        TAG_TREE_NODE_ID
      </tagId>
      [<name>
        TAG_TREE_NODE_NAME
      </name>]
      [<parentId>
        TAG_TREE_NODE_PARENT_ID
      </parentId>]
      [<position>
        TAG_TREE_NODE_POSITION
      </position>]
    </tag>]
  </room>
</rooms>
```



```
...
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</rooms>
```

2.1.3. Update a Household

It updates the information associated to an already existing household.

URL `https://vodkatv/external/admin/configuration/UpdateRoom.do`

Parameters POST parameter with the new information of the household identified by ROOM_ID in the following XML format:

```
<?xml version="1.0" encoding="UTF-8"?>
<room>
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    ROOM_DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
    </tagId>
  </tag>]
</room>
```

Response XML with the updated household information:

```
<?xml version="1.0" encoding="UTF-8"?>
<room>
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    ROOM_DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
    </tagId>
    [<name>
      TAG_TREE_NODE_NAME
    </name>]
  </tag>]
</room>
```

```
[<parentId>
    TAG_TREE_NODE_PARENT_ID
</parentId>]
[<position>
    TAG_TREE_NODE_POSITION
</position>]
</tag>]
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</room>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
not_found	<ul style="list-style-type: none">roomId	The household with identifier roomId does not exist on the system.

2.1.4. Delete a Household

It deletes an existing household and all their associated devices from the system.

URL <https://vodkatv/external/admin/configuration/DeleteRoom.do>

Parameters Multivalued GET parameter roomId with the household identifiers to be deleted:

?roomId=ROOM_ID_1&roomId=ROOM_ID_2

Response XML with the deleted room identifiers:

```
<?xml version="1.0" encoding="UTF-8"?>
<rooms>
    <room>
        <roomId>
            ROOM_ID_1
        </roomId>
    </room>
    <room>
        <roomId>
            ROOM_ID_2
        </roomId>
    </room>
    ...
<!-- This element is included only if an error occurs -->
<errors>
```

```
...
</errors>
</rooms>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
required	<ul style="list-style-type: none">roomId	No roomId was supplied.
not_found	<ul style="list-style-type: none">roomId	The household with identifier roomId does not exist on the system.

2.2. Devices

A VoDKA.TV device represents a physical device that accesses the VoDKA.TV system. Every created device must be associated with an existing household.

A device has the following attributes:

- **id** (INTEGER): internal device identifier. This value is autogenerated by the system. It must not be entered by the user when creating a device.
- **physicalId** (TEXT): the device physical identifier. This value must be the MAC address of the device. This field is case-insensitive, therefore there is no difference between uppercase letters and lowercase letters. It is important to take into account that if the MAC address is used to identify the devices, the format of the MAC must be exactly the same that the device provides. Thus, in Hansun or HED (IP12/Web:TV) set-top boxes this value is returned as xx:xx:xx:xx:xx:xx, and therefore it is mandatory to include the colons (:) between hexadecimal pairs.
- **deviceClass**: device class used to know more details about the device. The valid classes can be obtained using the operation described in section ??.

At the time of writing, the available types are:

- STBHED
- IPHONE
- SMARTPHONE
- MOBILEPHONE
- TABLET
- PC
- OTHER

Note that device classes are different from the device types (section ??). While the device class is a static value associated to the device when it is created, the device type is used to associate a device class with a user agent pattern. Thus, using the user agent VoDKA.TV can associate a device class to those devices that access the VoDKA.TV system, but are not provisioned, such as PCs or Tablets.

- `roomId` (TEXT(50)): household identifier which will have associated the device.
- `description` (TEXT): device description (optional).
- `tag`: the node of the tag tree to which this device is associated (see ??)(optional).
 - `tagId` (INTEGER): the identifier of the node of the tag tree.
- `version` (INTEGER): version number of the firmware installed on that device.
- `lastUpdateTime` (yyyy-MM-dd' T' HH:mm:ssZ): last time when the VoDKA.TV updates the information stored about the device.

2.2.1. Create a Device

It creates a new device with the specified values.

URL `https://vodkatv/external/admin/configuration/CreateDevice.do`

Parameters POST parameter with the information of the new device in the following XML format:

```
<?xml version="1.0" encoding="UTF-8"?>
<device>
  <physicalId>
    PHYSICAL_ID
  </physicalId>
  [<deviceClass>
    DEVICE_CLASS
  </deviceClass>]
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
    </tagId>
  </tag>]
</device>
```

Response XML with the new entered device information, including the generated internal identifier for the new device:

```
<?xml version="1.0" encoding="UTF-8"?>
<device>
  <id>
    ID
  </id>
  <physicalId>
    PHYSICAL_ID
  </physicalId>
  [<deviceClass>
    DEVICE_CLASS
  </deviceClass>]
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
    </tagId>
    [<name>
      TAG_TREE_NODE_NAME
    </name>]
    [<parentId>
      TAG_TREE_NODE_PARENT_ID
    </parentId>]
    [<position>
      TAG_TREE_NODE_POSITION
    </position>]
  </tag>]
  [<version>
    VERSION
  </version>]
  [<lastUpdateTime>
    LAST_UPDATE_TIME
  </lastUpdateTime>]
  <!-- This element is included only if an error occurs -->
  <errors>
    ...
  </errors>
</device>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
duplicated	<ul style="list-style-type: none">physicalId	A device with the same identifier physicalId already exists on the system.

2.2.2. Find a Device

It returns the device with the given identifier.

URL `https://vodkatv/external/admin/configuration/FindDeviceById.do`

Parameters GET parameter `deviceId` with the device identifier to search the device:

`?deviceId=DEVICE_ID`

Response XML specifying the device of the specified identifier:

```
<?xml version="1.0" encoding="UTF-8"?>
<device>
  <id>
    ID
  </id>
  <physicalId>
    PHYSICAL_ID
  </physicalId>
  [<deviceClass>
    DEVICE_CLASS
  </deviceClass>]
  <roomId>
    ROOM_ID
  </roomId>
  [<description>
    DESCRIPTION
  </description>]
  [<tag>
    <tagId>
      TAG_TREE_NODE_ID
    </tagId>
    [<name>
      TAG_TREE_NODE_NAME
    </name>]
    [<parentId>
      TAG_TREE_NODE_PARENT_ID
    </parentId>]
    [<position>
      TAG_TREE_NODE_POSITION
    </position>]
```

```
</tag>]
[<version>
  VERSION
</version>]
[<lastUpdateTime>
  LAST_UPDATE_TIME
</lastUpdateTime>]
<!-- This element is included only if an error occurs -->
<errors>
  ...
</errors>
</device>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
required	<ul style="list-style-type: none">deviceId	No deviceId was supplied.
not_found	<ul style="list-style-type: none">deviceId	The device with identifier deviceId does not exist on the system.
invalid_value	<ul style="list-style-type: none">deviceId	deviceId must be an integer value.

2.2.3. Find Devices

It returns the list of devices filtering by a given criteria.

URL <https://vodkatv/external/admin/configuration/FindDevices.do>

Parameters GET parameters :

- **startIndex**: index of the first device returned starting with 1.
- **count**: number of device returned.
- **sortBy**: field to sort the result (optional).
- **order**: type of order, ascending or descending (optional).
- **query**: string to search (optional).

?startIndex=START_INDEX&count=COUNT
&sortBy=SORT_BY&order=ORDER&query=QUERY

Response XML specifying the devices:

```
<?xml version="1.0" encoding="UTF-8"?>
<devices existsMore="TRUE|FALSE" countTotal="COUNT_TOTAL">
  <device>
    <id>
      ID
    </id>
    <physicalId>
      PHYSICAL_ID
    </physicalId>
    [<deviceClass>
      DEVICE_CLASS
    </deviceClass>]
    <roomId>
      ROOM_ID
    </roomId>
    [<description>
      DESCRIPTION
    </description>]
    [<tag>
      <tagId>
        TAG_TREE_NODE_ID
      </tagId>
      [<name>
        TAG_TREE_NODE_NAME
      </name>]
      [<parentId>
        TAG_TREE_NODE_PARENT_ID
      </parentId>]
      [<position>
        TAG_TREE_NODE_POSITION
      </position>]
    </tag>]
    [<version>
      VERSION
    </version>]
    [<lastUpdateTime>
      LAST_UPDATE_TIME
    </lastUpdateTime>]
  </device>
  <device>
    ...
  </device>
  ...
  <!-- This element is included only if an error occurs -->
  <errors>
    ...
  </errors>
</devices>
```


Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
required	<ul style="list-style-type: none">startIndex	No startIndex was supplied.
required	<ul style="list-style-type: none">count	No count was supplied.
invalid_value	<ul style="list-style-type: none">startIndex	startIndex must be an integer value.
invalid_value	<ul style="list-style-type: none">count	count must be an integer value.

2.2.4. Find the Devices of a Household

It returns a list of the devices from the specified household.

URL <https://vodkatv/external/admin/configuration/FindDevicesByRoom.do>

Parameters GET parameter `roomId` with the household identifier to search the devices:

?roomId=ROOM_ID

Response XML specifying the devices of the specified household:

```
<?xml version="1.0" encoding="UTF-8"?>
<devices existsMore="TRUE|FALSE" countTotal="COUNT_TOTAL">
  <device>
    <id>
      ID
    </id>
    <physicalId>
      PHYSICAL_ID
    </physicalId>
    [<deviceClass>
      DEVICE_CLASS
    </deviceClass>]
    <roomId>
      ROOM_ID
    </roomId>
    [<description>
      DESCRIPTION
    </description>]
    [<tag>
      <tagId>
```

```

        TAG_TREE_NODE_ID
    </tagId>
    [<name>
        TAG_TREE_NODE_NAME
    </name>]
    [<parentId>
        TAG_TREE_NODE_PARENT_ID
    </parentId>]
    [<position>
        TAG_TREE_NODE_POSITION
    </position>]
</tag>]
[<version>
    VERSION
</version>]
[<lastUpdateTime>
    LAST_UPDATE_TIME
</lastUpdateTime>]
</device>
<device>
    ...
</device>
...
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</devices>

```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
required	<ul style="list-style-type: none"> roomId 	No roomId was supplied.

2.2.5. Find the Devices of a Tag Node

It returns the list of the devices with the specified tag node.

URL <https://vodkatv/external/admin/configuration/FindDevicesByTagId.do>

Parameters GET parameters :

- **startIndex**: index of the first device returned starting with 1.
- **count**: number of device returned.

- **sortBy**: field to sort the result (optional).
- **order**: type of order, ascending or descending (optional).
- **query**: string to search (optional).
- **tagId**: node identifier to search the devices.

```
?startIndex=START_INDEX&count=COUNT  
&sortBy=SORT_BY&order=ORDER&query=QUERY&tagId=TAG_TREE_NODE_ID
```

Response XML specifying the devices of the specified tag tree node:

```
<?xml version="1.0" encoding="UTF-8"?>  
<devices existsMore="TRUE|FALSE" countTotal="COUNT_TOTAL">  
  <device>  
    <id>  
      ID  
    </id>  
    <physicalId>  
      PHYSICAL_ID  
    </physicalId>  
    [<deviceClass>  
      DEVICE_CLASS  
    </deviceClass>]  
    <roomId>  
      ROOM_ID  
    </roomId>  
    [<description>  
      DESCRIPTION  
    </description>]  
    [<tag>  
      <tagId>  
        TAG_TREE_NODE_ID  
      </tagId>  
      [<name>  
        TAG_TREE_NODE_NAME  
      </name>]  
      [<parentId>  
        TAG_TREE_NODE_PARENT_ID  
      </parentId>]  
      [<position>  
        TAG_TREE_NODE_POSITION  
      </position>]  
    </tag>]  
    [<version>  
      VERSION  
    </version>]  
    [<lastUpdateTime>  
      LAST_UPDATE_TIME  
    </lastUpdateTime>]
```

```

</device>
<device>
    ...
</device>
...
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</devices>

```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
required	<ul style="list-style-type: none"> tagId 	No tagId was supplied.
required	<ul style="list-style-type: none"> startIndex 	No startIndex was supplied.
required	<ul style="list-style-type: none"> count 	No count was supplied.
invalid_value	<ul style="list-style-type: none"> tagId 	tagId must be an integer value.
invalid_value	<ul style="list-style-type: none"> startIndex 	startIndex must be an integer value.
invalid_value	<ul style="list-style-type: none"> count 	count must be an integer value.

2.2.6. Update a Device

It updates the information associated to an already existing device.

URL <https://vodkatv/external/admin/configuration/UpdateDevice.do>

Parameters POST parameter with the new information of the device identified by ID in the following XML format:

```

<?xml version="1.0" encoding="UTF-8"?>
<device>
  <id>
    ID
  </id>
  <physicalId>
    PHYSICAL_ID
  </physicalId>
</device>

```

```
</physicalId>
[<deviceClass>
    DEVICE_CLASS
</deviceClass>]
<roomId>
    ROOM_ID
</roomId>
[<description>
    DESCRIPTION
</description>]
[<tag>
    <tagId>
        TAG_TREE_NODE_ID
    </tagId>
</tag>]
</device>
```

If the `tag` element is not included, it will be assumed as a set to null value, and hence the device will stop being associated to a tree node.

Response XML with the updated device information:

```
<?xml version="1.0" encoding="UTF-8"?>
<device>
    <id>
        ID
    </id>
    <physicalId>
        PHYSICAL_ID
    </physicalId>
    [<deviceClass>
        DEVICE_CLASS
    </deviceClass>]
    <roomId>
        ROOM_ID
    </roomId>
    [<description>
        DESCRIPTION
    </description>]
    [<tag>
        <tagId>
            TAG_TREE_NODE_ID
        </tagId>
        [<name>
            TAG_TREE_NODE_NAME
        </name>]
        [<parentId>
            TAG_TREE_NODE_PARENT_ID
        </parentId>]
        [<position>
```

```
        TAG_TREE_NODE_POSITION
    </position>]
</tag>]
[<version>
    VERSION
</version>]
[<lastUpdateTime>
    LAST_UPDATE_TIME
</lastUpdateTime>]
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</device>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
not_found	<ul style="list-style-type: none">deviceId	The device with identifier deviceId does not exist on the system.
duplicated	<ul style="list-style-type: none">physicalId	A device with the same identifier physicalId already exists on the system.
invalid_value	<ul style="list-style-type: none">deviceId	deviceId must be an integer value.

2.2.7. Delete a Device

It deletes an existing device from the system.

URL <https://vodkatv/external/admin/configuration/DeleteDevice.do>

Parameters Multivalued GET parameter deviceId with the internal device identifiers to be deleted:

?id=ID_1&id=ID_2

Response XML with the deleted device identifiers:

```
<?xml version="1.0" encoding="UTF-8"?>
<devices>
  <device>
    <id>
      ID_1
```

```
        </id>
    </device>
<device>
    <id>
        ID_2
    </id>
</device>
...
<!-- This element is included only if an error occurs -->
<errors>
    ...
</errors>
</devices>
```

Errors The possible errors that can be returned by this operation are:

Error code	Parameters	Error description
not_found	<ul style="list-style-type: none">deviceId	The device with identifier deviceId does not exist on the system.
required	<ul style="list-style-type: none">deviceId	No deviceId was supplied.
invalid_value	<ul style="list-style-type: none">deviceId	deviceId must be an integer value.