

Shopping and other ticketing systems

These methods are supported on Shopping 5.2 and later (and 5.1 with hotfix Q12679).

We recommend using the Shopping portal for provisioning software, but if you already have established process in place and only want to use Shopping for its delivery features, we have provided the following methods.

- [GetAllApplicationsInfo](#) – returns a list of all shoppable applications
- [GetUsersInfo](#) – returns user information based on the user account or email address
- [GetMachineInfo](#) – returns machine information based on its domain and machine name
- [GetOrderStatus](#) – returns the status of the order based on the order reference
- [PlaceOrder](#) – places order in Shopping
- [ApprovePendingOrder](#) – approves a pending order. Only supports a single approver workflow
- [RejectPendingOrder](#) – rejects a pending order. Only supports a single approver workflow

When invoking the POST APIs, make sure that *UserAgent* for the request is set to mimic a browser's user agent, for example. *Mozilla/5.0 (Windows NT 6.2; WOW64; rv:19.0) Gecko/20100101 Firefox/19.0*

Add *"X-Requested-With"* header to the request with value set to *"XMLHttpRequest"* while invoking the POST APIs.

GetAllApplicationsInfo

Gets a list of all the applications in the system.

Type

GET

Usage

URL: `[ShoppingBaseUrl]/RequestItems/GetAllApplicationsInfo?responseFormat=xml`

Input Parameters

`responseFormat` – the default response is in JSON format but can be set to XML with `responseFormat=xml`

Returns

- Example JSON response format

```
[{"ApplicationId":1,"ApplicationName":"AppModel 2012","Cost":0.0000,"ApplicationRef":"","ApprovalRequired":true,"IsSms":true,"ApplicationCategories":[{"ApplicationCategoryId":-1,"ApplicationCategoryName":"Miscellaneous"}],"RentalDays":0,"CanBeRented":false}, {"ApplicationId":20057,"ApplicationName":"TestApp1","Cost":0.0000,"ApplicationRef":"123456","ApprovalRequired":true,"IsSms":true,"ApplicationCategories":[{"ApplicationCategoryId":3,"ApplicationCategoryName":"SMSApps"}, {"ApplicationCategoryId":4,"ApplicationCategoryName":"slm"}],"RentalDays":0,"CanBeRented":false}]
```

- Example XML response format

```

<ArrayOfApplicationInfo xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001
/XMLSchema-instance">
  <ApplicationInfo>
    <ApplicationId>1</ApplicationId>
    <ApplicationName>AppModel 2012</ApplicationName>
    <Cost>0.0000</Cost>
    <ApplicationRef/>
    <ApprovalRequired>true</ApprovalRequired>
    <IsSms>true</IsSms>
    <ApplicationCategories>
      <ApplicationCategory>
        <ApplicationCategoryId>-1</ApplicationCategoryId>
        <ApplicationCategoryName>Miscellaneous</ApplicationCategoryName>
      </ApplicationCategory>
    </ApplicationCategories>
    <RentalDays>0</RentalDays>
    <CanBeRented>false</CanBeRented>
  </ApplicationInfo>
  <ApplicationInfo>
    <ApplicationId>20057</ApplicationId>
    <ApplicationName>TestApp1</ApplicationName>
    <Cost>0.0000</Cost>
    <ApplicationRef>123456</ApplicationRef>
    <ApprovalRequired>true</ApprovalRequired>
    <IsSms>true</IsSms>
    <ApplicationCategories>
      <ApplicationCategory>
        <ApplicationCategoryId>3</ApplicationCategoryId>
        <ApplicationCategoryName>SMSApps</ApplicationCategoryName>
      </ApplicationCategory>
      <ApplicationCategory>
        <ApplicationCategoryId>4</ApplicationCategoryId>
        <ApplicationCategoryName>slm</ApplicationCategoryName>
      </ApplicationCategory>
    </ApplicationCategories>
    <RentalDays>0</RentalDays>
    <CanBeRented>false</CanBeRented>
  </ApplicationInfo>
</ArrayOfApplicationInfo>

```

GetUsersInfo

Gets the user id.

Type

GET

Usage

- To get user details with the `userAccount` and/or `emailAccount` parameter:

```

URL: <ShoppingBaseUrl>/Users/GetUsersInfo?
userAccount=<userAccount>&emailAccount=<emailAccount>&responseFormat=xml

```

You need only use the `userAccount` or `emailAccount` parameter to get the user id. If both parameters are provided, `userAccount` takes precedence.

or

- To get user details with the `emailAccount` parameter:

```

URL: <ShoppingBaseUrl>/Users/GetUsersFromEmail?emailAccount=<emailAddress>&responseFormat=xml

```

Input parameters

- `userAccount` – the user account
- `emailAccount` – the email address for the user
- `responseFormat` – the default response is in JSON format but can be set to XML with `responseFormat=xml`

Returns

- Returns the user id if one is found. Else returns an empty string.
- Example JSON response format

```
[{"FullName": "Administrator", "Id": 2, "ManagerAccount": "", "ManagerEmail": "", "ManagerFullName": "", "UserAccount": "DEV26\\Administrator", "UserEmail": "Administrator@DEV26.local", "UserLanguage": "en-US", "IsSccmUser": true}]
```

- Example XML response format

```
<ArrayOfUserInfo xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <UserInfo>
    <FullName>Administrator</FullName>
    <Id>2</Id>
    <ManagerAccount/>
    <ManagerEmail/>
    <ManagerFullName/>
    <UserAccount>DEV26\\Administrator</UserAccount>
    <UserEmail>Administrator@DEV26.local</UserEmail>
    <UserLanguage>en-US</UserLanguage>
    <IsSccmUser>true</IsSccmUser>
  </UserInfo>
</ArrayOfUserInfo>
```

GetMachineInfo

Gets information about the machine based its domain and machine name.

Type

GET

Usage

- URL: `<ShoppingBaseUrl>/Machines/GetMachineInfo?domainName=<domainName>&machineName=<machineName>&responseFormat=xml`

Input parameters

- `domainName` – the domain the machine is associated with
- `machineName` – name of the machine
- `responseFormat` – the default response is in JSON format but can be set to XML with `responseFormat=xml`

Returns

- Example JSON response format

```
[{"Id": 2, "Name": "DEV26-CLIENT03"}]
```

- Example XML response response format

```
<ArrayOfMachineInfo xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <MachineInfo>
    <Id>2</Id>
    <Name>DEV26-CLIENT03</Name>
  </MachineInfo>
</ArrayOfMachineInfo>
```

GetOrderStatus

Returns the status of a particular order.

Type

- GET

Usage

- URL: `<ShoppingBaseUrl>/Orders/GetOrderStatus?requestRef=<requestRef>&responseFormat=xml`

Input parameters

- `requestRef` – the request reference of the order
- `responseFormat` – the default response is in JSON format but can be set to XML with `responseFormat=xml`

Returns

- Example JSON response

```
{ "OrderStatus": "Order Placed" }
```

- Example XML response

```
<?xml version="1.0" encoding="utf-8"?>
<string>Order Placed</string>
```

PlaceOrder

Enables an order to be placed for a particular user and machine. The user and machine must already exist in the Shopping database for this method to succeed.

Type

- POST

Usage

- URL: `<ShoppingBaseUrl>/RequestItem/PlaceOrder`

Input parameters

- `applicationid` – the application id to be installed. Derived from `GetAllApplicationsInfo`
- `machineid` – the machine id the application is to be installed on. Derived from `GetMachineInfo`
- `userid` – the user id requesting the application. Derived from `GetUsersInfo`
- `comments` – optional parameter to capture comments from the user making the request

Returns

- Returns the request reference if the order is successfully placed; else an empty string

ApprovePendingOrder

Approves a pending order. This method only supports the single approver workflow.

Type

- POST

Usage

- URL: `<ShoppingBaseUrl>/Approver/ApprovePendingOrder`

Input parameters

- `processingId` – the processing id of the order pending approval. This parameter takes precedence over `requestRef` if present
- `requestRef` – the request reference of the order pending approval. Derived from `GetOrderStatus`
- `comments` – optional parameter to capture comments from the approver

Returns

- Boolean – 1 if the order is approved and 0 if it is declined

RejectPendingOrder

Rejects a pending order. This method only supports the single approver workflow.

Type

- POST

Usage

- URL: `<ShoppingBaseUrl>/Approver/RejectPendingOrder`

Input parameters

- `processingId` – the processing id of the order pending approval. This parameter takes precedence over `requestRef` if present
- `requestRef` – the request reference of the order pending approval. Derived from `GetOrderStatus`
- `comments` – optional parameter to capture comments from the approver

Returns

- Boolean – 1 if the order is rejected and 0 if it is not rejected