

MS-WEBSS Test Suite Specification

**Abstract:** This document provides information about how to configure the test suite and how the MS-WEBSS test suite is designed to test the MS-WEBSS Open Specification usability and accuracy. It describes test assumptions, scope and constraints of the test suite. It also specifies test scenarios, detail test cases, test suite architecture and adapter design.

Contents

[1 Configuring the test suite 5](#_Toc356294323)

[1.1 Configuring the test suite client 5](#_Toc356294324)

[1.1.1 Configuring the test suite client manually 5](#_Toc356294325)

[1.1.2 Configuring the test suite client by scripts 6](#_Toc356294326)

[1.2 Configuring the system under test (SUT) 6](#_Toc356294327)

[1.2.1 Configuring the SUT manually 6](#_Toc356294328)

[1.2.2 Configuring the SUT by scripts 6](#_Toc356294329)

[1.3 Configuring the SHOULD/MAY requirements 6](#_Toc356294330)

[2 Test suite design 7](#_Toc356294331)

[2.1 Assumptions, scope and constraints 7](#_Toc356294332)

[Assumptions 7](#_Toc356294333)

[Scope 7](#_Toc356294334)

[In scope 7](#_Toc356294335)

[Out of scope 7](#_Toc356294336)

[Constraints 7](#_Toc356294337)

[2.2 Test suite architecture 7](#_Toc356294338)

[2.3 Technical dependencies and encryption considerations 9](#_Toc356294339)

[Technical dependencies 9](#_Toc356294340)

[Encryption consideration 9](#_Toc356294341)

[2.4 Adapter design 9](#_Toc356294342)

[2.4.1 Adapter overview 9](#_Toc356294343)

[Protocol adapter 9](#_Toc356294344)

[SUT control adapters 9](#_Toc356294345)

[2.4.2 Technical feasibility of adapter approach 10](#_Toc356294346)

[Message generation 10](#_Toc356294347)

[Message consumption 10](#_Toc356294348)

[SUT control adapter 10](#_Toc356294349)

[2.4.3 Adapter abstract layer 10](#_Toc356294350)

[Protocol adapter 10](#_Toc356294351)

[MS-WEBSS adapter interface 10](#_Toc356294352)

[SUT control adapters 11](#_Toc356294353)

[2.4.4 Adapter details 11](#_Toc356294354)

[*2.4.4.1* Protocol adapter 11](#_Toc356294355)

[2.4.4.1.1 MS-WEBSS protocol adapter 11](#_Toc356294356)

[Adapter interface 12](#_Toc356294357)

[Adapter implementation 13](#_Toc356294358)

[Other class 13](#_Toc356294359)

[*2.4.4.2* SUT control adapters 13](#_Toc356294360)

[2.4.4.2.1 MS-WEBSS SUT control adapter 13](#_Toc356294361)

[2.5 Test scenarios 14](#_Toc356294362)

[2.5.1 S01\_OperationsOnContentType 14](#_Toc356294363)

[Description 14](#_Toc356294364)

[Operations 14](#_Toc356294365)

[Prerequisites 14](#_Toc356294366)

[Cleanup 14](#_Toc356294367)

[2.5.2 S02\_OperationsOnContentTypeXmlDocument 15](#_Toc356294368)

[Description 15](#_Toc356294369)

[Operations 15](#_Toc356294370)

[Prerequisites 15](#_Toc356294371)

[Cleanup 15](#_Toc356294372)

[2.5.3 S03\_OperationsOnPage 15](#_Toc356294373)

[Description 15](#_Toc356294374)

[Operations 15](#_Toc356294375)

[Prerequisites 15](#_Toc356294376)

[Cleanup 15](#_Toc356294377)

[2.5.4 S04\_OperationsOnFile 15](#_Toc356294378)

[Description 15](#_Toc356294379)

[Operations 15](#_Toc356294380)

[Prerequisites 15](#_Toc356294381)

[Cleanup 15](#_Toc356294382)

[2.5.5 S05\_OperationsOnObjectId 16](#_Toc356294383)

[Description 16](#_Toc356294384)

[Operations 16](#_Toc356294385)

[Prerequisites 16](#_Toc356294386)

[Cleanup 16](#_Toc356294387)

[2.5.6 S06\_OperationsOnListTemplates 16](#_Toc356294388)

[Description 16](#_Toc356294389)

[Operations 16](#_Toc356294390)

[Prerequisites 16](#_Toc356294391)

[Cleanup 16](#_Toc356294392)

[2.5.7 S07\_OperationsOnColumns 16](#_Toc356294393)

[Description 16](#_Toc356294394)

[Operations 16](#_Toc356294395)

[Prerequisites 16](#_Toc356294396)

[Cleanup 16](#_Toc356294397)

[2.5.8 S08\_OperationsOnCSS 17](#_Toc356294398)

[Description 17](#_Toc356294399)

[Operations 17](#_Toc356294400)

[Prerequisites 17](#_Toc356294401)

[Cleanup 17](#_Toc356294402)

[2.5.9 S09\_OperationsOnWeb 17](#_Toc356294403)

[Description 17](#_Toc356294404)

[Operations 17](#_Toc356294405)

[Prerequisites 17](#_Toc356294406)

[Cleanup 17](#_Toc356294407)

[2.5.10 S10\_OperationsOnActivatedFeatures 17](#_Toc356294408)

[Description 17](#_Toc356294409)

[Operations 17](#_Toc356294410)

[Prerequisites 17](#_Toc356294411)

[Cleanup 17](#_Toc356294412)

[2.6 Test case design 18](#_Toc356294413)

[2.6.1 Traditional test case design 18](#_Toc356294414)

[Test case selection 18](#_Toc356294415)

[2.6.2 Test case description 23](#_Toc356294416)

# Configuring the test suite

## Configuring the test suite client

### Configuring the test suite client manually

Before you run the test suite, update the values in the MS-SITESS\_TestSuite.deployment.ptfconfig file. The MS-WEBSS\_TestSuite.deployment.ptfconfig file can also be configured by running the client setup script.

1. Open MS-WEBSS\TestSuite\MS-WEBSS\_TestSuite.deployment.ptfconfig.
2. Update the following value to specify the common configuration file.

Property name= "CommonConfigurationFileName" value="SharePointCommonConfiguration.deployment.ptfconfig"

**Note** This property can be removed or set to empty if the required properties are copied to the test suite specific configuration file. Any other changes to this property will cause all test cases in the test suite to fail during execution. The test suite first search through its specific configuration file and use the properties from there if they are defined, before looking for them from the common configuration file (if specified).

1. Update the following properties' values to match SUT settings and configuration.

* Property name="TargetServiceUrl" value="[TransportType]://[SUTComputerName]/sites/[SiteCollectionName][EntryUrl]"
* Property name="SiteCollectionName" value="MSWEBSS\_SiteCollection"
* Property name="EntryUrl" value="/\_vti\_bin/webs.asmx"
* Property name="SiteCollectionUrl" value="[TransportType]://[SUTComputerName]/sites/[SiteCollectionName]"
* Property name="CreateContentType\_ValidateContentType" value="0x0101"/
* Property name="SubSiteUrl" value="[TransportType]://[SUTComputerName]/sites/[SiteCollectionName]/[webSiteName]"
* The following properties are not associated with SUT settings and can normally retain with default values.
* Property name="webSiteName" value="MSWEBSS\_Site"
* Property name="TestSiteTitle" value="MSWEBSS\_SiteTitle"
* Property name="TestSiteDescription" value="MSWEBSS\_SiteDescription"
* Property name="TestSiteLanguage" value="1033"
* Property name="FoldName" value="MSWEBSS\_DocumentLibrary"
* Property name="DocName" value="MSWEBSS\_TestData.docx"
* Property name="CssFile\_Valid" value="core.css"
* Property name="GetCustomizedPageStatus\_ValidFileUrl" value ="[SubSiteUrl]/default.aspx"
* Property name="GetObjectIdFromUrl\_ListUrl" value="[SubSiteUrl]/SitePages/Forms/AllPages.aspx"
* Property name="GetObjectIdFromUrl\_ListItemUrl" value="[SubSiteUrl]/SitePages/Home.aspx"
* Property name="GetObjectIdFromUrl\_NoListRelatedUrl" value="[SubSiteUrl]/default.aspx"
* Property name="RevertFileContentStream\_ValidFileUrl" value ="[SubSiteUrl]/default.aspx"
* Property name="UpdateColumns\_Type" value="Text"
* Property name ="WebUrlFromPageUrl\_PageUrl" value="[SubSiteUrl]/Shared%20Documents/Forms/AllItems.aspx"
* Property name="ServiceTimeOut" value="10"

### Configuring the test suite client by scripts

To configure the test suite client using scripts, see section 5.2.4 of the [SharePointTestSuiteDeploymentGuide.docx](../SharePointTestSuiteDeploymentGuide.docx).

## Configuring the system under test (SUT)

### Configuring the SUT manually

To manually configure the SUT, see section 5.1.3 of the [SharePointTestSuiteDeploymentGuide.docx](../SharePointTestSuiteDeploymentGuide.docx).

### Configuring the SUT by scripts

To configure the SUT using scripts, see section 5.1.2 of the [SharePointTestSuiteDeploymentGuide.docx](../SharePointTestSuiteDeploymentGuide.docx).

## Configuring the SHOULD/MAY requirements

Implementation of the SHOULD/MAY and endnote-related requirements are pre-configured in the format "<Property name="RXXXEnabled" value="XXXX"/>" for three product versions in the following config files:

* MS-WEBSS\_WindowsSharePointServiceS03\_SHOULDMAY.deployment.ptfconfig
* MS-WEBSS\_SharePointServeMS-WEBSS\_R2007\_SHOULDMAY.deployment.ptfconfig
* MS-WEBSS\_SharePointFoundation2010\_SHOULDMAY.deployment.ptfconfig
* MS-WEBSS\_SharePointServeMS-WEBSS\_R2010\_SHOULDMAY.deployment.ptfconfig
* MS-WEBSS\_SharePointFoundation2013\_SHOULDMAY.deployment.ptfconfig
* MS-WEBSS\_SharePointServeMS-WEBSS\_R2013\_SHOULDMAY.deployment.ptfconfig

If RXXXEnabled is set to true, the requirement must be checked. If false, the requirement must not be checked. For Microsoft product versions, all values should not be changed. For third-party products, the closest Microsoft product version should be chosen, and the value of RXXXEnabled should be updated according to the real product behavior. For example, if choosing SharePoint Foundation 2010, open **MS-WEBSS\_SharePointFoundation2010\_SHOULDMAY.deployment.ptfconfig** and update the RXXXEnabled accordingly.

# Test suite design

## Assumptions, scope and constraints

Assumptions

* None

Scope

In scope

* This test suite will verify the accuracy and integrity of the technical content in the Open Specification against the results returned from the protocol server by using four operations: GetVersions, DeleteAllVersions, DeleteVersion and RestoreVersion.
* This test suite will verify the Full WSDL which is provided in the Open Specification.
* This test suite will verify the server-side and testable requirements by running all the test cases on both HTTP and HTTPS.
* This test suite will verify operations over SOAP 1.1 and SOAP 1.2.

Out of scope

* This test suite will not verify the requirements related to client behaviors.
* This test suite will not verify the requirements related to server internal behaviors.
* This test suite will not verify the internal implementations of its transport protocol stack.

Constraints

* None

## Test suite architecture

This test suite verifies the server-side and testable requirements obtained from the Open Specification. The following figure shows the architecture of this test suite.

 **The architecture of the test suite**

The details of the MS-WEBSS test suite architecture

* SUT hosts the MS-WEBSS Web Service which this test suite runs against.
* From third-party user’s point of view, SUT is the protocol server implementation.
* The following products have been tested with the MS-WEBSS test suite on Windows platform.
* Windows SharePoint Services 3.0 Service Pack 3 (SP3)
* Microsoft SharePoint Foundation 2010 Service Pack 2 (SP2)
* Microsoft SharePoint Foundation 2013 SP1
* Microsoft Office SharePoint Server 2007 Service Pack 3 (SP3)
* Microsoft SharePoint Server 2010 Service Pack 2 (SP2)
* Microsoft SharePoint Server 2013 SP1
* Test suite acts as the client to communicate with the SUT and Valids the requirements gathered from MS-WEBSS Open Specification.
* The test cases invoke the MS-WEBSS Adapter to call the MS-WEBSS operations and Valid the response from SUT. Test cases also use the SUT control adapter to set the SUT to the test case specific situation.
* MS-WEBSS adapter is used in the test cases. The test cases call the methods in the adapter interfaces to invoke the MS-WEBSS operation.
* The test cases also use the SUT control adapter to set/modify the SUT environment by calling the methods in the SUT control adapter interface to configure the SUT.

## Technical dependencies and encryption considerations

Technical dependencies

* This test suite depends on the SOAP messaging protocol for exchanging structured data and type information.
* This test suite depends on HTTP protocol or HTTPS protocol to transmit the messages.
* This test suite depends on the wsdl.exe tool in the .NET Framework SDK to generate the MS-WEBSS proxy class.
* This test suite depends on Protocol Test Framework (PTF) to derive managed adapters.

Encryption consideration

* Transportation of MS-WEBSS includes HTTP and HTTPS, and encryption will be handled by HTTPS.

## Adapter design

### Adapter overview

One protocol adapter and one SUT control adapter will be designed for this test suite.

Protocol adapter

* MS-WEBSS adapter
* The MS-WEBSS adapter is a managed adapter, which is derived from the ManagedAdapterBase class in PTF.
* The MS-WEBSS adapter has the following functionalities
* Choose HTTP or HTTPS and SOAP 1.1 or 1.2 for transport;
* Construct requests of 21 MS-WEBSS operations;
* Communicate with the SUT by sending requests to the SUT and receive the corresponding responses from the SUT;
* Parse the response messages and validate the messages according to the WSDL schema;
* Generate the result log.
* The MS-WEBSS adapter uses the C# proxy class, which is generated by running the wsdl.exe tool against the full WSDL of this protocol to send SOAP request messages and receive SOAP response messages. The wsdl.exe can be found in Microsoft .NET Framework SDK tools.

SUT control adapters

* The MS-WEBSS SUT control adapter uses PowerShell script implementation.
* The MS-WEBSS SUT control adapter has the following functionalities.
* Set content type property to Read-only or Sealed.
* Get the object IDs from list or list item or active features.
* The SUT control adapter is invoked by the test cases.

### Technical feasibility of adapter approach

Message generation

The MS-WEBSS adapter gets the parameter values of the WSDL operations and calls the corresponding operations in MS-WEBSS proxy class, the MS-WEBSS proxy class serializes the parameter values to XML elements to format the SOAP request messages, then the SOAP request messages are sent out by the MS-WEBSS proxy class.

Message consumption

The messages received from the SUT will be parsed in the MS-WEBSS proxy class and be passed upon to the MS-WEBSS adapter. Then these messages are consumed in the MS-WEBSS adapter to validate the message format and to validate the logic-related requirements in the test cases.

SUT control adapter

The SUT control adapter is designed to remotely control the SUT to set content type property, get object ID.

### Adapter abstract layer

Protocol adapter

MS-WEBSS adapter interface

There are 22 methods declared in the MS-WEBSS adapter interface IMS\_WEBSSAdapter.

|  |  |  |
| --- | --- | --- |
| No. | Methods | Description |
| 1 | CreateContentType | Used to create a new content type on the context site |
| 2 | CustomizeCss | Used to enable customization of the specified cascading style sheet (CSS) for the context site. |
| 3 | DeleteContentType | Used remove a given content type from the site. |
| 4 | GetActivatedFeatures | Used to get a list of activated features on the site and on the parent site collection. |
| 5 | GetAllSubWebCollection | Used to get a list of the titles and URLs of all sites in the site collection. |
| 6 | GetColumns | Used to get the collection of column definitions for all the columns available on the context site. |
| 7 | GetContentType | Used to get the collection of column definitions for all the columns available on the context site. |
| 8 | GetContentTypes | Used to retrieve all content types for a given context site. |
| 9 | GetCustomizedPageStatus | Used to obtain properties of the object referenced by the specified URL. |
| 10 | GetListTemplates | Used to get the customization status (also known as the ghosted status) of the specified page or file. |
| 11 | GetObjectIdFromUrl | Used to get the collection of list template definitions for the context site. |
| 12 | GetWeb | Used to get the Title, URL, Description, Language, and theme properties of the specified site. |
| 13 | GetWebCollection | Used to get the Title and URL properties of all immediate child sites of the context site. |
| 14 | InitializeService | Initialize the services of MS-WEBSS with the specified transport type, soap version and user authentication provided by the test case. |
| 15 | RemoveContentTypeXmlDocument | Used to remove an XML document in the XML document collection of a site content type. |
| 16 | RevertAllFileContentStreams | Used to revert all pages within the context site to their original states. |
| 17 | RevertCss | Used to revert the customizations of the context site defined by the given CSS file and return those customizations to the default style. |
| 18 | RevertFileContentStream | Used to revert the specified page within the context site to its original state. |
| 19 | UpdateColumns | Used to perform the following operation on the context site and all child sites within its hierarchy.  Add one or more specified new columns.  Update one or more specified existing columns.  Delete one or more specified existing columns. |
| 20 | UpdateContentType | Used to update a content type on the context site. |
| 21 | UpdateContentTypeXmlDocument | Used to add or update an XML document in the XML Document collection of a site content type. |
| 22 | WebUrlFromPageUrl | Used to get the URL of the parent site of the specified URL. |

SUT control adapters

* MS-WEBSS SUT control adapter interface
* Two methods defined in SUT control adapter interface ISUTControlAdapter.
* There are two methods declared in the LISTSWS SUT control adapter interface IMS\_WEBSSSUTControlAdapter.

|  |  |  |
| --- | --- | --- |
| No. | Methods | Description |
| 1 | GetObjectId | Get the Object Id(s) from list or listItem or active feature of the web site. |
| 2 | SetContentTypeReadOnlyOrSealed | Used to set ContentType property to Read-Only. |

### Adapter details

#### Protocol adapter

##### MS-WEBSS protocol adapter

The following figure shows the class diagram of the MS-WEBSS adapter.



Protocol adapter class diagram

The following outlines details of the class diagram:Adapter interface

* IMS\_WEBSSAdapter is the interface of the protocol adapter.
* IMS\_WEBSSAdapter defines the 22 methods invoked by test cases. See the details of the 22 methods in section [2.4.3 Adpter Abstract Layer](#_Adapter_abstract_layer).

Adapter implementation

* MS\_WEBSSAdapter is protocol adapter which implements the method which defined in the interface IMS-WEBSSAdapter. It performs 21 WSDL operations and initializes web service and test cases.
* The Initialize method is used to initialize the MS-WEBSS test suite.

Other class

* WebSoap is the proxy class created from full WSDL of the Open Specification. It is used by MS-WEBSSAdapter to invoke the 21 operations to communicate with SUT by sending/receiving the SOAP messages.
* The AdapterHelper provides methods to assist MS-WEBSSAdapter.

#### SUT control adapters

##### MS-WEBSS SUT control adapter

The following figure shows the class diagram of the SUT control adapter.



SUT control adapter class diagram

* The following outlines details of the class diagram:

IMS\_WEBSSSUTControlAdapter is interface of SUT control adapter class which defined all the methods in SUT control adapter class.

## Test scenarios

Ten scenarios are designed to cover the server-side, testable requirements in the MS-WEBSS test suite. The details of the scenarios are as follows

|  |  |
| --- | --- |
| Scenario | Description |
| S01\_OperationsOnContentType | Protocol client tries to perform operations associated with content types. |
| S02\_OperationsOnContentTypeXmlDocument | Protocol client tries to perform operations associated with XML document. |
| S03\_OperationsOnPage | Protocol client tries to perform operations associated with page. |
| S04\_OperationsOnFile | Protocol client tries to perform operations associated with file. |
| S05\_OperationsOnObjectId | Protocol client tries to perform operations associated with objectId. |
| S06\_OperationsOnListTemplates | Protocol client tries to get the collection of list templates definitions. |
| S07\_OperationsOnColumns | Protocol client tries to perform operations associated with columns. |
| S08\_OperationsOnCSS | Protocol client tries to perform operations associated with the customization of the specified CSS. |
| S09\_OperationsOnWeb | Protocol client tries to perform operations associated with sub-webs, webs and web collection. |
| S10\_OperationsOnActivatedFeatures | Protocol client tries to perform operations associated with activated features. |

MS-WEBSS Scenarios

### S01\_OperationsOnContentType

Description

Protocol client tries to perform operations associated with content types.

Operations

* CreateContentType
* GetContentType
* UpdateContentType
* DeleteContentType

Prerequisites

N/A

Cleanup

N/A

### S02\_OperationsOnContentTypeXmlDocument

Description

Protocol client tries to perform operations associated with XML document.

Operations

* RemoveContentTypeXmlDocument
* UpdateContentTypeXmlDocument

Prerequisites

N/A

Cleanup

N/A

### S03\_OperationsOnPage

Description

Protocol client tries to perform operations associated with page.

Operations

* GetCustomizedPageStatus
* WebUrlFromPageUrl

Prerequisites

N/A

Cleanup

N/A

### S04\_OperationsOnFile

Description

Protocol client tries to perform operations associated with file.

Operations

* RevertFileContentStream
* RevertAllFileContentStreams

Prerequisites

N/A

Cleanup

N/A

### S05\_OperationsOnObjectId

Description

Protocol client tries to perform operations associated with objectId.

Operations

GetObjectIdFromUrl

Prerequisites

N/A

Cleanup

N/A

### S06\_OperationsOnListTemplates

Description

Protocol client tries to get the collection of list templates definitions.

Operations

GetListTemplates

Prerequisites

N/A

Cleanup

N/A

### S07\_OperationsOnColumns

Description

Protocol client tries to perform operations associated with columns.

Operations

* GetColumns
* UpdateColumns

Prerequisites

N/A

Cleanup

N/A

### S08\_OperationsOnCSS

Description

Protocol client tries to perform operations associated with the customization of the specified CSS.

Operations

* CustomizeCss
* RevertCss

Prerequisites

N/A

Cleanup

N/A

### S09\_OperationsOnWeb

Description

Protocol client tries to perform operations associated with sub-webs, webs and web collection.

Operations

* GetWeb
* GetWebCollection
* GetAllSubWebCollection

Prerequisites

N/A

Cleanup

N/A

### S10\_OperationsOnActivatedFeatures

Description

Protocol client tries to perform operations associated with activated features.

Operations

GetActivatedFeatures

Prerequisites

N/A

Cleanup

N/A

## Test case design

### Traditional test case design

Traditional testing approach is selected as the test approach for this test suite. 113 test cases are designed to cover the server-side and testable requirements.

Test case selection

113 traditional test cases are designed to cover the ten scenarios mentioned in section [2.5 Test scenarios](#_Test_Scenarios). Details of the traditional test cases are specified in section [2.6.2 Test case sescription](#_Test_Cases_Description). The scenarios distributions of the test cases are listed in the following table.

|  |  |
| --- | --- |
| Scenario ID | Test case name |
| S01\_OperationsOnContentType | [MSWEBSS\_S01\_TC01\_CreateContentType](#MSWEBSS_S01_TC01) |
| [MSWEBSS\_S01\_TC02\_DeleteContentType](#MSWEBSS_S01_TC02) |
| [MSWEBSS\_S01\_TC03\_GetContentTypesWithElements](#MSWEBSS_S01_TC03) |
| [MSWEBSS\_S01\_TC04\_GetContentTypes](#MSWEBSS_S01_TC04) |
| [MSWEBSS\_S01\_TC05\_UpdateContentTypeWithInvalidDeleteField](#MSWEBSS_S01_TC05) |
| [MSWEBSS\_S01\_TC06\_UpdateContentTypeFoundTheField](#MSWEBSS_S01_TC06) |
| [MSWEBSS\_S01\_TC07\_CreateContentTypeInvalidContentTypeId](#MSWEBSS_S01_TC07) |
| [MSWEBSS\_S01\_TC08\_CreateContentTypeInvalidDisplayNameGreaterThan128Characters](#MSWEBSS_S01_TC08) |
| [MSWEBSS\_S01\_TC09\_CreateContentTypeInvalidDisplayNameBeginsWithWhiteSpace](#MSWEBSS_S01_TC09) |
| [MSWEBSS\_S01\_TC10\_CreateContentTypeInvalidDisplayNameEndWithPeriod](#MSWEBSS_S01_TC10) |
| [MSWEBSS\_S01\_TC11\_CreateContentTypeInvalidDisplayNameEndWithSpace](#MSWEBSS_S01_TC11) |
| [MSWEBSS\_S01\_TC12\_CreateContentTypeInvalidBackslash](#MSWEBSS_S01_TC12) |
| [MSWEBSS\_S01\_TC13\_CreateContentTypeInvalidSlash](#MSWEBSS_S01_TC13) |
| [MSWEBSS\_S01\_TC14\_CreateContentTypeInvalidColon](#MSWEBSS_S01_TC14) |
| [MSWEBSS\_S01\_TC15\_CreateContentTypeInvalidAsterisk](#MSWEBSS_S01_TC15) |
| [MSWEBSS\_S01\_TC16\_CreateContentTypeInvalidQuestionMark](#MSWEBSS_S01_TC16) |
| [MSWEBSS\_S01\_TC17\_CreateContentTypeInvalidQuotationMark](#MSWEBSS_S01_TC17) |
| [MSWEBSS\_S01\_TC18\_CreateContentTypeInvalidNumberSign](#MSWEBSS_S01_TC18) |
| [MSWEBSS\_S01\_TC19\_CreateContentTypeInvalidPercentSign](#MSWEBSS_S01_TC19) |
| [MSWEBSS\_S01\_TC20\_CreateContentTypeInvalidLessThanSign](#MSWEBSS_S01_TC20) |
| [MSWEBSS\_S01\_TC21\_CreateContentTypeInvalidGreaterThanSign](#MSWEBSS_S01_TC21) |
| [MSWEBSS\_S01\_TC22\_CreateContentTypeInvalidOpeningCurlyBraces](#MSWEBSS_S01_TC22) |
| [MSWEBSS\_S01\_TC23\_CreateContentTypeInvalidClosingCurlyBraces](#MSWEBSS_S01_TC23) |
| [MSWEBSS\_S01\_TC24\_CreateContentTypeInvalidAmpersand](#MSWEBSS_S01_TC24) |
| [MSWEBSS\_S01\_TC25\_CreateContentTypeInvalidWithPeriod](#MSWEBSS_S01_TC25) |
| [MSWEBSS\_S01\_TC26\_CreateContentTypeInvalidDoublePeriods](#MSWEBSS_S01_TC26) |
| [MSWEBSS\_S01\_TC27\_CreateContentTypeInvalidTab](#MSWEBSS_S01_TC27) |
| [MSWEBSS\_S01\_TC28\_CreateContentTypeInvalidVerticalBar](#MSWEBSS_S01_TC28) |
| [MSWEBSS\_S01\_TC29\_GetContentTypeInvalidWithNull](#MSWEBSS_S01_TC29) |
| [MSWEBSS\_S01\_TC30\_GetContentTypeInvalidWithEmpty](#MSWEBSS_S01_TC30) |
| [MSWEBSS\_S01\_TC31\_GetContentTypeInvalidateWithInvalidContentType](#MSWEBSS_S01_TC31) |
| [MSWEBSS\_S01\_TC32\_UpdateContentTypeInvalidContentTypeId](#MSWEBSS_S01_TC32) |
| [MSWEBSS\_S01\_TC33\_CreateContentTypeWithSameName](#MSWEBSS_S01_TC33) |
| [MSWEBSS\_S01\_TC34\_UpdateContentTypeWithShouldMay](#MSWEBSS_S01_TC34) |
| [MSWEBSS\_S01\_TC35\_DeleteContentTypeInvalidIdentifier](#MSWEBSS_S01_TC35) |
| [MSWEBSS\_S01\_TC36\_DeleteContentTypeReadOnly](#MSWEBSS_S01_TC36) |
| [MSWEBSS\_S01\_TC37\_CreateContentType\_Unauthenticated](#MSWEBSS_S01_TC37) |
| [MSWEBSS\_S01\_TC38\_DeleteContentType\_Unauthenticated](#MSWEBSS_S01_TC38) |
| [MSWEBSS\_S01\_TC39\_GetContentType\_Unauthenticated](#MSWEBSS_S01_TC39) |
| [MSWEBSS\_S01\_TC40\_GetContentTypes\_Unauthenticated](#MSWEBSS_S01_TC40) |
| [MSWEBSS\_S01\_TC41\_UpdateContentType\_Unauthenticated](#MSWEBSS_S01_TC41) |
| [MSWEBSS\_S01\_TC42\_CreateContentTypeInvalidParentType](#MSWEBSS_S01_TC42) |
| [MSWEBSS\_S01\_TC43\_CreateContentTypeAddAndUpdateNewFields](#MSWEBSS_S01_TC43) |
| [MSWEBSS\_S01\_TC44\_UpdateContentTypeWithValidContentType](#MSWEBSS_S01_TC44) |
| [MSWEBSS\_S01\_TC45\_DeleteContentTypeWithValidContentType](#MSWEBSS_S01_TC45) |
| [MSWEBSS\_S01\_TC46\_GetContentTypesWithShouldMay](#MSWEBSS_S01_TC46) |
| S02\_OperationsOnContentTypeXmlDocument | [MSWEBSS\_S02\_TC01\_RemoveContentTypeXmlDocumentInvalidWithEmpty](#MSWEBSS_S02_TC01) |
| [MSWEBSS\_S02\_TC02\_UpdateContentTypeXmlDocument\_Unauthenticated](#MSWEBSS_S02_TC02) |
| [MSWEBSS\_S02\_TC03\_UpdateContentTypeXmlDocumentInvalidContentType](#MSWEBSS_S02_TC03) |
| [MSWEBSS\_S02\_TC04\_UpdateContentTypeXmlDocumentInvalidXmlElement](#MSWEBSS_S02_TC04) |
| [MSWEBSS\_S02\_TC05\_RemoveContentTypeXmlDocument\_Unauthenticated](#MSWEBSS_S02_TC05) |
| [MSWEBSS\_S02\_TC06\_RemoveContentTypeXmlDocument](#MSWEBSS_S02_TC06) |
| [MSWEBSS\_S02\_TC07\_UpdateContentTypeXmlDocument](#MSWEBSS_S02_TC07) |
| S03\_OperationsOnPage | [MSWEBSS\_S03\_TC01\_GetCustomizedPageStatus\_ValidFileUrl](#MSWEBSS_S03_TC01) |
| [MSWEBSS\_S03\_TC02\_GetCustomizedPageStatus\_InvalidFileUrl](#MSWEBSS_S03_TC02) |
| [MSWEBSS\_S03\_TC03\_WebUrlFromPageUrl](#MSWEBSS_S03_TC03) |
| [MSWEBSS\_S03\_TC04\_GetCustomizedPageStatus\_Unauthenticated](#MSWEBSS_S03_TC04) |
| [MSWEBSS\_S03\_TC05\_WebUrlFromPageUrl\_Unauthenticated](#MSWEBSS_S03_TC05) |
| S04\_OperationsOnFile | [MSWEBSS\_S04\_TC01\_RevertAllFileContentStreams](#MSWEBSS_S04_TC01) |
| [MSWEBSS\_S04\_TC02\_RevertFileContentStream\_ValidFileUrl](#MSWEBSS_S04_TC02) |
| [MSWEBSS\_S04\_TC03\_RevertFileContentStream\_InvalidPageUrl](#MSWEBSS_S04_TC03) |
| [MSWEBSS\_S04\_TC04\_RevertFileContentStream\_InvalidSiteUrl](#MSWEBSS_S04_TC04) |
| [MSWEBSS\_S04\_TC05\_RevertFileContentStream\_ValidUrlReferToParentSite](#MSWEBSS_S04_TC05) |
| [MSWEBSS\_S04\_TC06\_RevertFileContentStream\_Empty](#MSWEBSS_S04_TC06) |
| [MSWEBSS\_S04\_TC07\_RevertFileContentStream\_NULL](#MSWEBSS_S04_TC07) |
| [MSWEBSS\_S04\_TC08\_RevertFileContentStream\_NotPartOfSiteDefinition](#MSWEBSS_S04_TC08) |
| [MSWEBSS\_S04\_TC09\_RevertFileContentStream\_NotReferToPage](#MSWEBSS_S04_TC09) |
| [MSWEBSS\_S04\_TC10\_RevertAllFileContentStreams\_Unauthenticated](#MSWEBSS_S04_TC10) |
| [MSWEBSS\_S04\_TC11\_RevertFileContentStream\_Unauthenticated](#MSWEBSS_S04_TC11) |
| S05\_OperationsOnObjectId | [MSWEBSS\_S05\_TC01\_GetObjectIdFromUrl\_ValidListUrl](#MSWEBSS_S05_TC01) |
| [MSWEBSS\_S05\_TC02\_GetObjectIdFromUrl\_ValidListItemUrl](#MSWEBSS_S05_TC02) |
| [MSWEBSS\_S05\_TC03\_GetObjectIdFromUrl\_NoListOrListItemUrl](#MSWEBSS_S05_TC03) |
| [MSWEBSS\_S05\_TC04\_GetObjectIdFromUrl\_InvalidUrl](#MSWEBSS_S05_TC04) |
| [MSWEBSS\_S05\_TC05\_GetObjectIdFromUrl\_Unauthenticated](#MSWEBSS_S05_TC05) |
| S06\_OperationsOnListTemplates | [MSWEBSS\_S06\_TC01\_GetListTemplates\_Unauthenticated](#MSWEBSS_S06_TC01) |
| [MSWEBSS\_S06\_TC02\_GetListTemplates\_Succeed](#MSWEBSS_S06_TC02) |
| S07\_OperationsOnColumns | [MSWEBSS\_S07\_TC01\_UpdateColumns\_InvalidFieldDefinition](#MSWEBSS_S07_TC01) |
| [MSWEBSS\_S07\_TC02\_GetColumns\_WithInvalidColumn](#MSWEBSS_S07_TC02) |
| [MSWEBSS\_S07\_TC03\_UpdateColumns\_AllFieldsValid](#MSWEBSS_S07_TC03) |
| [MSWEBSS\_S07\_TC04\_UpdateColumns\_NoMatchingName](#MSWEBSS_S07_TC04) |
| [MSWEBSS\_S07\_TC05\_GetColumns\_Unauthenticated](#MSWEBSS_S07_TC05) |
| [MSWEBSS\_S07\_TC06\_UpdateColumns\_Unauthenticated](#MSWEBSS_S07_TC06) |
| [MSWEBSS\_S07\_TC07\_UpdateColumns\_NewFieldsWithMultipleMethodNoFields](#MSWEBSS_S07_TC07) |
| [MSWEBSS\_S07\_TC08\_UpdateColumns\_UpdateFieldWithMultipleMethodNoFields](#MSWEBSS_S07_TC08) |
| [MSWEBSS\_S07\_TC09\_UpdateColumns\_DeleteFieldsWithMultipleMethodNoFields](#MSWEBSS_S07_TC09) |
| [MSWEBSS\_S07\_TC10\_UpdateColumns\_ExistingColumn](#MSWEBSS_S07_TC10) |
| [MSWEBSS\_S07\_TC11\_UpdateColumns\_NonexistentColumn](#MSWEBSS_S07_TC11) |
| [MSWEBSS\_S07\_TC12\_UpdateColumns\_InvalidGUID](#MSWEBSS_S07_TC12) |
| [MSWEBSS\_S07\_TC13\_UpdateColumns\_NoNameOrDisplayName](#MSWEBSS_S07_TC13) |
| S08\_OperationsOnCSS | [MSWEBSS\_S08\_TC01\_CustomizeCss\_Unauthenticated](#MSWEBSS_S08_TC01) |
| [MSWEBSS\_S08\_TC02\_RevertCss\_CssFileNotExist](#MSWEBSS_S08_TC02) |
| [MSWEBSS\_S08\_TC03\_CustomizeCss\_CssFileBlankOrMissing](#MSWEBSS_S08_TC03) |
| [MSWEBSS\_S08\_TC04\_CustomizeCss\_NoFileExtension](#MSWEBSS_S08_TC04) |
| [MSWEBSS\_S08\_TC05\_CustomizeCss\_CssFileNotExist](#MSWEBSS_S08_TC05) |
| [MSWEBSS\_S08\_TC06\_CustomizeCss\_ContainAsterisk](#MSWEBSS_S08_TC06) |
| [MSWEBSS\_S08\_TC07\_RevertCss\_Unauthenticated](#MSWEBSS_S08_TC07) |
| [MSWEBSS\_S08\_TC08\_RevertCss\_Succeed](#MSWEBSS_S08_TC08) |
| [MSWEBSS\_S08\_TC09\_RevertCss\_CssFileBlankOrNull](#MSWEBSS_S08_TC09) |
| [MSWEBSS\_S08\_TC10\_CustomizeCssFileMatchName](#MSWEBSS_S08_TC10) |
| S09\_OperationsOnWeb | [MSWEBSS\_S09\_TC01\_GetWeb\_Unauthenticated](#MSWEBSS_S09_TC01) |
| [MSWEBSS\_S09\_TC02\_GetWeb\_Succeed](#MSWEBSS_S09_TC02) |
| [MSWEBSS\_S09\_TC03\_GetWeb\_InvalidUrl](#MSWEBSS_S09_TC03) |
| [MSWEBSS\_S09\_TC04\_GetWeb\_NoWebUrl](#MSWEBSS_S09_TC04) |
| [MSWEBSS\_S09\_TC05\_GetWebCollection\_Unauthenticated](#MSWEBSS_S09_TC05) |
| [MSWEBSS\_S09\_TC06\_GetWebCollection\_Succeed](#MSWEBSS_S09_TC06) |
| [MSWEBSS\_S09\_TC07\_GetAllSubWebCollection\_Unauthenticated](#MSWEBSS_S09_TC07) |
| [MSWEBSS\_S09\_TC08\_GetAllSubWebCollection\_Succeed](#MSWEBSS_S09_TC08) |
| [MSWEBSS\_S09\_TC09\_GetWebDefinition](#MSWEBSS_S09_TC09) |
| [MSWEBSS\_S09\_TC10\_GetWebResponseWithValidXml](#MSWEBSS_S09_TC10) |
| S10\_OperationsOnActivatedFeatures | [MSWEBSS\_S10\_TC01\_GetActivatedFeaturesAboveWSS3](#MSWEBSS_S10_TC01) |
| [MSWEBSS\_S10\_TC02\_GetActivatedFeatures\_Succeed](#MSWEBSS_S10_TC02) |
| [MSWEBSS\_S10\_TC03\_GetActivatedFeatures\_Unauthenticated](#MSWEBSS_S10_TC03) |
| [MSWEBSS\_S10\_TC04\_GetActivatedFeaturesResponseWithCommaDelimitedGUID](#MSWEBSS_S10_TC04) |

Test case scenario distribution

### Test case description

There are 113 traditional test cases designed in this test suite. Common prerequisites and Common cleanup for all the test cases are

|  |  |  |
| --- | --- | --- |
| Common prerequisites | Common HTTP Prerequisites | Client uses HTTP protocol and authenticated user account. |
| Common HTTPS Prerequisites | Client uses HTTPS protocol and authenticated user account. |
| Common cleanup | N/A | |

* The steps in the following test case definitions use methods and parameters in the adapter interfaces directly.
* All the test cases will run on HTTP except MSWEBSS\_S10\_TC02\_GetActivatedFeatures\_Succeed will run on HTTPS as a sample testing, which is a positive case.

The following tables describe the traditional test case.

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC01\_CreateContentType |
| Description | This test case aims to verify create a new content type on the context site with valid parameters. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request with valid parameters.   **Input parameters:**  dispflayName: Valid   1. The server will respond with the CreateContentTypeResult containing created content type identifier. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC01\_CreateContentType

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC02\_DeleteContentType |
| Description | This test case aims to verify remove a given content type from the site with valid parameters. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a DeleteContentType request with valid parameter.   **Input parameters**:  contentTypeId: Valid   1. The server will respond with DeleteContentTypeResult stating that the content type has been removed from the site. 2. The result object returned in step 3 is checked for presence of contentTypeid. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC02\_DeleteContentType

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC03\_GetContentTypesWithElements |
| Description | This test case aims to verify the GetContentTypes operation, and the response should contain all content types on a site. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetContentTypes request with valid parameters.   **Input parameters**:  contentTypeId: valid   1. If the server respond with the ContentType Data.   The result object returned in step 3 is checked for whether the content type has localization.   1. The result object returned in step 3 is checked for the content type requested has the featureId attribute set to a non-empty GUID. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC03\_GetContentTypesWithElements

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC04\_GetContentTypes |
| Description | This test case aims to verify GetContentTypes operation with valid content in the response. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetContentTypes request.   **Input parameters:**  No parameter.  The server will respond with all ContentTypes. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC04\_GetContentTypes

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC05\_UpdateContentTypeWithInvalidDeleteField |
| Description | This test case aims to verify UpdateContentType operation with invalid delete field. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentType request.   **Input parameters:**  contentTypeID: valid  newProperty: valid  newFields: valid  updateFields: null  deleteField: valid   1. The server will respond with an UpdateContentTypeResult. 2. The result object returned in step 3 is checked for if a reference to the field is found on the content type. If the field is not found on the content type, the value of ErrorCode MUST be nonzero, |
| Cleanup | N/A |

MSWEBSS\_S01\_TC05\_UpdateContentTypeWithInValidDeleteField

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC06\_UpdateContentTypeFoundTheField |
| Description | This test case aims to verify the UpdateContentType operation when the updateFields can be found. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentType request.   **Input parameters:**  contentTypeID: valid  newProperty: valid  newFields: valid  updateFields: valid  deleteField: valid   1. The server will respond with an UpdateContentTypeResult. 2. The result object returned in step 3 is checked for the results of content type. If data of results is not null,   If methods element including the ID,   1. The result object returned in step 3 is checked for ErrorCode. If the errorcode is set to 0x00000000. 2. The result object returned in step 3 is checked for the localization of content type. If it has localization, ListProperties.Locs] value MUST be set to 1. 3. The result object returned in step 3 is checked for the content type is sealed. If it is sealed, ListProperties.Sealed value MUST be set to 1. 4. The result object returned in step 3 is checked for the ListProperties of content type. |
| Cleanup | N/A |

MSWEBSS\_S01\_ TC06\_UpdateContentTypeFoundTheFiled

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC07\_CreateContentTypeInvalidContentTypeId |
| Description | This test case aims to verify CreateContentType operation with invalid parameters. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Valid  contentTypeID: Specify a invalid content type identifier or is not the content type identifier of a content type that is available on the context site  newFields: valid  contentTypeType: valid   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC07\_CreateContentTypeInvalidContentTypeId

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC08\_CreateContentTypeInvalidDisplayNameGreaterThan128Characters |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName longer than 128 characters. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name longer than 128 characters.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

[MSWEBSS\_S01\_TC08\_CreateContentTypeInvalidDisplayNameGreaterThan128Characters](#MSWEBSS_S01_TC08)

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC09\_CreateContentTypeInvalidDisplayNameBeginsWithWhiteSpace |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName beginning with whitespace. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name beginning with whitespace.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC09\_CreateContentTypeInvalidDisplanyNameBeginsWithWhitespace

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC10\_CreateContentTypeInvalidDisplayNameEndWithPeriod |
| Description | This test case aims to verify the the CreateContentType operation with invalid displayName ending with a period. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name ending with a period.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC10\_CreateContentTypeInvalidDisplayNameEndWithPeriod

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC11\_CreateContentTypeInvalidDisplayNameEndWithSpace |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName ending with whitespace. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name ending with whitespace.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC11\_CreateContentTypeInvalidDisplayNameEndWithSpace

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC12\_CreateContentTypeInvalidBackslash |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName with backslash. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name with backslash.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC12\_CreateContentTypeInvalidBackslash

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC13\_CreateContentTypeInvalidSlash |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName with slash. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name with slash.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC13\_CreateContentTypeInvalidSlash

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC14\_CreateContentTypeInvalidColon |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName with colon. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | * + 1. Common steps of authorized user.     2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name with colon.   * + 1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC14\_CreateContentTypeInvalidColon

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC15\_CreateContentTypeInvalidAsterisk |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains asterisk. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains asterisk.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC15\_CreateContentTypeInvalidAsterisk

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC16\_CreateContentTypeInvalidQuestionMark |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains question mark. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains question mark.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC16\_CreateContentTypeInvalidQuestionMark

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC17\_CreateContentTypeInvalidQuotationMark |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains quotation mark. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains quotation mark.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC17\_CreateContentTypeInvalidQuotationMark

|  |  |
| --- | --- |
| **S01\_OperationsOnContentType** | |
| **Test case ID** | MSWEBSS\_S01\_TC18\_CreateContentTypeInvalidNumberSign |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains number sign. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains number sign.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC18\_CreateContentTypeInvalidNumberSign

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC19\_CreateContentTypeInvalidPercentSign |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains percent sign. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains percent sign.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC19\_CreateContentTypeInvalidPercentSign

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC20\_CreateContentTypeInvalidLessThanSign |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains less than sign. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common stpes. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains less than sign.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC20\_CreateContentTypeInvalidLessThanSign

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC21\_CreateContentTypeInvalidGreaterThanSign |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains great than sign. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains great than sign.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC21\_CreateContentTypeInvalidGreaterThanSign

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC22\_CreateContentTypeInvalidOpeningCurlyBraces |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains opening curly braces. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains opening curly braces.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC22\_CreateContentTypeInvalidOpeningCurlyBraces

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC23\_CreateContentTypeInvalidClosingCurlyBraces |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains ClosingCurlyBraces. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains ClosingCurlyBraces.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC23\_CreateContentTypeInvalidClosingCurlyBraces

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC24\_CreateContentTypeInvalidAmpersand |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains Ampersand. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains Ampersand.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC24\_CreateContentTypeInvalidAmpersand

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC25\_CreateContentTypeInvalidWithPeriod |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName with period. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name with period.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC25\_CreateContentTypeInvalidWithPeriod

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC26\_CreateContentTypeInvalidDoublePeriods |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName with double periods. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common stpes. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name with double periods.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC26\_CreateContentTypeInvalidDoublePeriods

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC27\_CreateContentTypeInvalidTab |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains tab. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CreateContentType request.   **Input parameters:**  displayName: Specify a name contains tab.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC27\_CreateContentTypeInvalidTab

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC28\_CreateContentTypeInvalidVerticalBar |
| Description | This test case aims to verify the CreateContentType operation with invalid displayName contains verticalBar. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user.  2. Send a CreateContentType request.  **Input parameters:**  displayName: Specify a name contains verticalBar.  3. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC28\_CreateContentTypeInvalidVerticalBar

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC29\_GetContentTypeInvalidWithNull |
| Description | This test case aims to verify the GetContentType operation with invalid contentTypeId which is a null content type ID. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetContentType request.   **Input parameters:**  contentTypeID: null.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC29\_GetContentTypeInvalidWithNull

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC30\_GetContentTypeInvalidWithEmpty |
| Description | This test case aims to verify the GetContentType operation with invalid contentTypeId which is an empty content type ID. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetContentType request.   **Input parameters:**  contentTypeID: Specify an empty content type ID   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC30\_GetContentTypeInvalidWithEmpty

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC31\_GetContentTypeInvalidateWithInvalidContentType |
| Description | This test case aims to verify the GetContentType operation with invalid contentTypeId which does not match the content type ID of a content type on the site. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetContentType request.   **Input parameters:**  contentTypeID: Specify a content type ID which does not match the content type ID of a content type on the site.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC31\_GetContentTypeInvalidateWithInvalidContentType

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC32\_UpdateContentTypeInvalidContentTypeId |
| Description | This test case aims to verify the UpdateContentType operation with invalid contentTypeId. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentType request.   **Input parameters:**  contentTypeID: Specify an invalid content type ID.  newProperties: Valid  newFields: Valid  updateFields: Valid  deleteFields: Valid   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC32\_UpdateContentTypeInvalidContentTypeId

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC33\_CreateContentTypeWithSameName |
| Description | This test case aims to verify CreateContentType operation with invalid displayName that has existed or used. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Try to create a new content type with a display name that has used by other existed content type. 2. Assert an exception is returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC33\_ CreateContentTypeWithSameName

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC34\_UpdateContentTypeWithShouldMay |
| Description | This test case aims to verify the optional behaviors in GetContentType operation. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentType request.   **Input parameters:**  contentTypeID: valid  newProperty: valid  newFields: valid  updateFields: valid  deleteField: valid   1. The server will respond with an UpdateContentTypeResult. If the server responds with the UpdateContentTypeResult containing content type identifier in Windows SharePoint Services 3.0 and above follow this behavior. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC34\_UpdateContentTypeWithShouldMay

|  |  |
| --- | --- |
| **S01\_OperationsOnContentType** | |
| Test case ID | MSWEBSS\_S01\_TC35\_DeleteContentTypeInvalidIdentifier |
| Description | This test case aims to verify the DeleteContentType operation with invalid content type ID. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user.   Input Parameters:  contentTypeId: invalid   1. Send a DeleteContentType request. 2. A SOAP exception will be returned from the server. 3. The exception returned in step 3 is checked for the contenTtypeId. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC35\_DeleteContentTypeInvalidIdentifier

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC36\_DeleteContentTypeReadOnly |
| Description | This test case aims to verify the DeleteContentType operation with valid contentTypeId which is a Read-Only content type. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Call SetContentTypeReadOnly method in SUT control adapter set content type to Read-Only. 3. Send a DeleteContentType request.   **Input parameters:**  contentTypeId: specify a content type ID whose content type is set to Read-only.   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC36\_DeleteContentTypeReadOnly

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC37\_CreateContentType\_Unauthenticated |
| Description | This test case is used to test CreateContentType operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a CreateContentType request with valid parameters.   **Input parameters:**  displayName: Valid   1. The server will return an HTTP 401 error when unauthorized user sends CreateContentType request. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC37\_CreateContentType\_Unauthenticated

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC38\_DeleteContentType\_Unauthenticated |
| Description | This test case is used to test DeleteContentType operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a DeleteContentType request.   **Input parameters:**  contentTypeId: valid   1. The server will return an HTTP 401 error when unauthorized user sends DeleteContentType request. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC38\_DeleteContentType\_Unauthenticated

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC39\_GetContentType\_Unauthenticated |
| Description | This test case aims to verify the GetContentType operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetContentType request.   **Input parameters:**  contentTypeId: valid   1. The server will return an HTTP 401 error when unauthorized user sends GetContentType request. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC39\_GetContentType\_Unauthenticated

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC40\_GetContentTypes\_Unauthenticated |
| Description | This test case aims to verify the GetContentTypes operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetContentTypes request.   **Input parameters:**  None   1. The server will return an HTTP 401 error when unauthorized user sends GetContentTypes request. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC40\_GetContentTypes\_Unauthenticated

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC41\_UpdateContentType\_Unauthenticated |
| Description | This test case aims to verify the UpdateContentTypes operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetContentTypes request.   **Input parameters:**  contentTypeID: valid  newProperty: valid  newFields: valid  updateFields: valid  deleteField: valid   1. The server will return an HTTP 401 error when unauthorized user sends UpdateContentTypes request. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC41\_UpdateContentType\_Unauthenticated

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC42\_CreateContentTypeInvalidParentType |
| Description | This test case aims to verify CreateContentType operation with invalid parent type. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **CreateContentType** request.   **Input parameters:**  displayName: Valid  contentTypeID: Specify an invalid content type identifier that is available on the context site.  newFields: invalid  contentTypeId: valid   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC42\_CreateContentTypeInvalidParentType

|  |  |
| --- | --- |
| **S01\_OperationsOnContentType** | |
| Test case ID | MSWEBSS\_S01\_TC43\_CreateContentTypeAddAndUpdateNewFields |
| Description | This test case aims to verify CreateContentType operation with invalid parameters. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **CreateContentType** request.   **Input parameters:**  displayName: Valid  contentTypeID: Specify an invalid content type identifier that is available on the context site.  newFields: add the valid parameter.  contentTypeId: update the valid value   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC43\_CreateContentTypeAddAndUpdateNewFields

|  |  |
| --- | --- |
| **S01\_OperationsOnContentType** | |
| Test case ID | MSWEBSS\_S01\_TC44\_UpdateContentTypeWithValidContentType |
| Description | This test case aims to verify the operation UpdateContentType on the context site with valid parameters. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **GetContentType** request.   **Input parameters:**  contentTypeId: valid   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC44\_UpdateContentTypeWithValidContentType

|  |  |
| --- | --- |
| S01\_OperationsOnContentType | |
| Test case ID | MSWEBSS\_S01\_TC45\_DeleteContentTypeWithValidContentType |
| Description | This test case aims to verify the DeleteContentType operation with valid contentTypeId. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **DeleteContentType** request.   **Input parameters:**  contentTypeId: valid   1. The result will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S01\_TC45\_DeleteContentTypeWithValidContentType

|  |  |
| --- | --- |
| **S01\_OperationsOnContentType** | |
| Test case ID | MSWEBSS\_S01\_TC46\_GetContentTypesWithShouldMay |
| Description | This test case aims to verify the optional behaviors of GetContentTypes operation. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **GetContentTypes** request with valid parameters.   **Input Parameters:**  None.   1. If the server responds with the GetContentTypesResult containing content type identifier in Windows SharePoint Services 3.0 and above follow this behavior, |
| Cleanup | N/A |

MSWEBSS\_S01\_TC46\_GetContentTypesWithShouldMay

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC01\_RemoveContentTypeXmlDocumentInvalidWithEmpty |
| Description | This test case aims to verify the RemoveContentTypeXmlDocument operation with invalid content type ID. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RemoveContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: Specify a content type ID that is not found on the site.  documenturl: valid   1. A SOAP exception will be returned from the server. 2. The exception returned in step 3 is checked for the contenTtypeId. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC01\_RemoveContentTypeXmlDocumentInvalidWithEmpty

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC02\_UpdateContentTypeXmlDocument\_Unauthenticated |
| Description | This test case aims to verify the UpdateContentTypeXmlDocument operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send an UpdateContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: valid  newDocument: valid   1. The server will return an HTTP 401 error when unauthorized user sends UpdateContentTypeXmlDocument request. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC02\_UpdateContentTypeXmlDocument\_Unauthenticated

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC03\_UpdateContentTypeXmlDocumentInvalidContentType |
| Description | This test case aims to verify the UpdateContentTypeXmlDocument operation with invalid contentTypeId. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: specify a content type ID that cannot be found  newDocument: valid   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC03\_UpdateContentTypeXmlDocumentInvalidContentType

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC04\_UpdateContentTypeXmlDocumentInvalidXmlElement |
| Description | This test case aims to verify the UpdateContentTypeXmlDocument operation with invalid newDocument value. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: valid  newDocument: invalid   1. A SOAP exception will be returned from the server. 2. The exception returned in step 3 is checked for the newDocument. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC04\_UpdateContentTypeXmlDocumentInvalidXmlElement

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC05\_RemoveContentTypeXmlDocument\_Unauthenticated |
| Description | This test case aims to verify the RemoveContentTypesXmlDocument operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send an UpdateContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: valid  newDocument: valid  The server will return an HTTP 401 error when unauthorized use sends UpdateContentTypesXmlDocument request. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC05\_RemoveContentTypeXmlDocument\_Unauthenticated

|  |  |
| --- | --- |
| S02\_OperationsOnContentTypeXmlDocument | |
| Test case ID | MSWEBSS\_S02\_TC06\_RemoveContentTypeXmlDocument |
| Description | This test case aims to verify RemoveContentTypeXmlDocument operation to remove a document from the document collection of a site content type. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RemoveContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: valid  documenturl: valid   1. A RemoveContentTypeXmlDocumentResponse will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC06\_RemoveContentTypeXmlDocument

|  |  |
| --- | --- |
| **S02\_OperationsOnContentTypeXmlDocument** | |
| Test case ID | MSWEBSS\_S02\_TC07\_UpdateContentTypeXmlDocument |
| Description | This test case aims to verify UpdateContentTypeXmlDocument operation to update an XML document in the XML Document collection of a site content type. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateContentTypeXmlDocument request.   **Input parameters:**  contentTypeId: valid  newDocument: valid   1. An UpdateContentTypeXmlDocumentResponse will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S02\_TC07\_UpdateContentTypeXmlDocument

|  |  |
| --- | --- |
| S03\_OperationsOnPage | |
| Test case ID | MSWEBSS\_S03\_TC01\_GetCustomizedPageStatus\_ValidFileUrl |
| Description | This test case aims to verify GetCustomizedPageStatus operation (also known as the ghosted status) of the specified page or file. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetCustomizedPageStatus request.   **Input parameters:**  fileUrl: valid   1. The server returns a customization status value which specifies whether the fileUrl parameter represents a customized page, an uncustomized page, or a file containing static content. 2. Send a GetCustomizedPageStatus request.   **Input parameters:**  fileUrl: static HTML file. |
| Cleanup | N/A |

MSWEBSS\_S03\_TC01\_GetCustomizedPageStatus\_ValidFileUrl

|  |  |
| --- | --- |
| **S03\_OperationsOnPage** | |
| Test case ID | MSWEBSS\_S03\_TC02\_GetCustomizedPageStatus\_InvalidFileUrl |
| Description | This test case aims to verify the GetCustomizedPageStatus operation with invalid parameter. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetCustomizedPageStatus request.   **Input parameters:**  fileUrl: invalid   1. A SOAP fault containing error code 0x80070002 will be returned from the server.   The result object returned in step 3 is checked for the error code 0x80070002 will be returned when relative to the protocol server address; the value of fileUrl does not specify a valid page or file.   1. Send a GetCustomizedPageStatus request.   **Input parameters:**  fileUrl: invalid, in which server portion is not an address of a site. |
| Cleanup | N/A |

MSWEBSS\_S03\_TC02\_GetCustomizedPageStatus\_InvalidFileUrl

|  |  |
| --- | --- |
| S03\_OperationsOnPage | |
| Test case ID | MSWEBSS\_S03\_TC03\_WebUrlFromPageUrl |
| Description | This test case aims to verify WebUrlFromPageUrl operation to get the URL of the parent site of the specified URL. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a WebUrlFromPageUrl request.   **Input parameters:**  pageUrl: valid   1. The full URL of the site will be returned from the server. If it does, |
| Cleanup | N/A |

MSWEBSS\_S03\_TC03\_WebUrlFromPageUrl

|  |  |
| --- | --- |
| S03\_OperationsOnPage | |
| Test case ID | MSWEBSS\_S03\_TC04\_GetCustomizedPageStatus\_Unauthenticated |
| Description | This test case aims to verify the GetCustomizedPageStatus operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetCustomizedPageStatus request.   **Input parameters:**  fileUrl: valid   1. The server will return an HTTP 401 error. |
| Cleanup | N/A |

MSWEBSS\_S03\_TC04\_GetCustomizedPageStatus\_Unauthenticated

|  |  |
| --- | --- |
| S03\_OperationsOnPage | |
| Test case ID | MSWEBSS\_S03\_TC05\_WebUrlFromPageUrl\_Unauthenticated |
| Description | This test case is used to test WebUrlFromPageUrl operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a WebUrlFromPageUrl request.   **Input parameters:**  pageUrl: valid   1. The server will return an HTTP 401 error. |
| Cleanup | N/A |

MSWEBSS\_S03\_TC05\_WebUrlFromPageUrl\_Unauthenticated

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC01\_RevertAllFileContentStreams |
| Description | This test case aims to verify RevertAllFileContentStreams operation to let all pages within the context site revert to their original states. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertAllFileContentStreams request.   **Input parameters:**  None parameter.  A RevertAllFileContentStreamsReponse XML element will be returned from sever. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC01\_RevertAllFileContentStreams

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC02\_RevertFileContentStream\_ValidFileUrl |
| Description | This test case aims to verify RevertFileContentStream operation to let the specified page within the context site revert to its original state. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: valid.  A RevertFileContentStreamResponse XML element will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC02\_RevertFileContentStream\_ValidFileUrl

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC03\_RevertFileContentStream\_InvalidPageUrl |
| Description | This test case aims to verify the RevertFileContentStream operation with invalid parameter. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: invalid URL for the page   1. A SOAP fault will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC03\_RevertFileContentStream\_InvalidPageUrl

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC04\_RevertFileContentStream\_InvalidSiteUrl |
| Description | This test case aims to verify the RevertFileContentStream operation with invalid parameter. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: Specifies an invalid site.   1. A SOAP fault will be returned from the server when the site referred by the fileUrl is not a valid site. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC04\_RevertFileContentStream\_InvalidSiteUrl

|  |  |
| --- | --- |
| **S04\_OperationsOnFile** | |
| Test case ID | MSWEBSS\_S04\_TC05\_RevertFileContentStream\_ValidUrlReferToParentSite |
| Description | This test case aims to verify the RevertFileContentStream operation with valid fileUrl but refers to a page on the parent site of the context site. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: Specifies a valid URL but refers to a page on the parent site of the context site.   1. A SOAP fault will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC05\_RevertFileContentStream\_ValidUrlReferToParentSite

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC06\_RevertFileContentStream\_Empty |
| Description | This test case aims to verify the **RevertFileContentStream** operation with valid fileUrl which is an empty URL. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: empty   1. A SOAP fault contains error code 0x82000001 will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC06\_RevertFileContentStream\_Empty

|  |  |
| --- | --- |
| **S04\_OperationsOnFile** | |
| Test case ID | MSWEBSS\_S04\_TC07\_RevertFileContentStream\_NULL |
| Description | This test case aims to verify the RevertFileContentStream operation with invalid fileUrl which is NULL. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: NULL   1. A SOAP fault contains error code 0x82000001 will be returned from the server. 2. The result object returned in step 3 is checked for the appropriate error code will be returned when there is any problem in performing the operation. 3. The result object returned in step 3 is checked for the error code 0x82000001 will be returned when an empty URL is passed in or no fileUrl element is supplied. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC07\_RevertFileContentStream\_NULL

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC08\_RevertFileContentStream\_NotPartOfSiteDefinition |
| Description | This test case aims to verify the RevertFileContentStream operation with valid fileUrl which is not part of site definition. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: Specifies a site that is not part of the site definition.   1. A SOAP fault contains error code 0x80131600 will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC08\_RevertFileContentStream\_NotPartOfSiteDefinition

|  |  |
| --- | --- |
| **S04\_OperationsOnFile** | |
| Test case ID | MSWEBSS\_S04\_TC09\_RevertFileContentStream\_NotReferToPage |
| Description | This test case aims to verify the RevertFileContentStream operation with valid fileUrl which is not referring to page on the content site. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: Specifies a valid URL but does not referring to a page on the context site.   1. A SOAP fault contains error code 0x80007002 will be returned from the server. 2. The result object returned in step 3 is checked for the error code 0x80007002 will be returned when a valid URL is passed in that does NOT refer to a page on the context site. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC09\_RevertFileContentStream\_NotReferToPage

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC10\_RevertAllFileContentStreams\_Unauthenticated |
| Description | This test case aims to verify RevertAllFileContentStreams when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a RevertAllFileContentStreams request with valid parameters.   **Input parameters:**  None   1. The server will return an HTTP 401 error when unauthorized user sends RevertAllFileContentStreams request. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC10\_RevertAllFileContentStreams\_Unauthenticated

|  |  |
| --- | --- |
| S04\_OperationsOnFile | |
| Test case ID | MSWEBSS\_S04\_TC11\_RevertFileContentStream\_Unauthenticated |
| Description | This test case is used to test RevertFileContentStream operation when the user is not authorized. |
| Prerequisites | Common HTTP Prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a RevertFileContentStreams request.   **Input parameters:**  fileUrl: valid   1. The server will return an HTTP 401 error when unauthorized user sends RevertFileContentStreams request. |
| Cleanup | N/A |

MSWEBSS\_S04\_TC11\_RevertFileContentStream\_Unauthenticated

|  |  |
| --- | --- |
| **S05\_OperationsOnObjectId** | |
| Test case ID | MSWEBSS\_S05\_TC01\_GetObjectIdFromUrl\_ValidListUrl |
| Description | This test case aims to verify the GetObjectIdFromUrl operation when the objectUrl specifies a list. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Check whether the environment is Windows SharePoint Services 3.0   Valid the checking result.   1. If Yes, exit. 2. If No, execute the next step. 3. Send a GetObjectIdFromUrl request.   **Input parameters:**  objectUrl: specify a list.   1. A GetObjectIdFromUrlResponse object with listId and ListServerTemplate, ListBaseType will be returned from server. 2. Call getObjectId method in SUT Control Adapter to get objectId from GetObjectIdFromUrlResult if the object is a list, the value of the attribute MUST be the list identifier. |
| Cleanup | N/A |

MSWEBSS\_S05\_TC01\_GetObjectIdFromUrl\_ValidListUrl

|  |  |
| --- | --- |
| **S05\_OperationsOnObjectId** | |
| Test case ID | MSWEBSS\_S05\_TC02\_GetObjectIdFromUrl\_ValidListItemUrl |
| Description | This test case aims to verify the GetObjectIdFromUrl operation when the objectUrl specifies a list item. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Check whether the environment is Windows SharePoint Services 3.0   Valid the checking result.   1. If Yes, exit. 2. If No, execute the next step. 3. Send a GetObjectIdFromUrl request.   **Input parameters:**  objectUrl: specify a list item.   1. A GetObjectIdFromUrlResponse object with listItemId will be returned from server. |
| Cleanup | N/A |

MSWEBSS\_S05\_TC02\_GetObjectIdFromUrl\_ValidListItemUrl

|  |  |
| --- | --- |
| S05\_OperationsOnObjectId | |
| Test case ID | MSWEBSS\_S05\_TC03\_GetObjectIdFromUrl\_NoListOrListItemUrl |
| Description | This test case aims to verify the GetObjectIdFromUrl operation when the objectUrl does not specifies a list or a list item. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Check whether the environment is Windows SharePoint Services 3.0   Valid the checking result.   1. If Yes, exit. 2. If No, execute the next step. 3. Send a GetObjectIdFromUrl request.   **Input parameters:**  objectUrl: doesn’t specify a list or a list item.   1. Only a GetObjectIdFromUrlResponseGetObjectIdFromUrlResult oebject without any elements will be returned from server. |
| Cleanup | N/A |

MSWEBSS\_S05\_TC03\_GetObjectIdFromUrl\_NoListOrListItemUrl

|  |  |
| --- | --- |
| **S05\_OperationsOnObjectId** | |
| Test case ID | MSWEBSS\_S05\_TC04\_GetObjectIdFromUrl\_InvalidUrl |
| Description | This test case aims to verify the GetObjectIdFromUrl operation with invalid objectUrl. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Check whether the environment is Windows SharePoint Services 3.0   Valid the checking result.   1. If Yes, exit. 2. If No, execute the next step. 3. Send a GetObjectIdFromUrl request.   **Input parameters:**  objectUrl: specify URL which is not corresponds to an object on the site   1. A SOAP exception will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S05\_TC04\_GetObjectIdFromUrl\_InvalidUrl

|  |  |
| --- | --- |
| S05\_OperationsOnObjectId | |
| Test case ID | MSWEBSS\_S05\_TC05\_GetObjectIdFromUrl\_Unauthenticated |
| Description | This test case aims to verify the GetObjectIdFromUrl operation when the user is unauthenticated. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetObjectIdFromUrl request.   **Input parameters:**  objectUrl: valid   1. The server will return an HTTP 401 error when authorized user sends GetObjectIdFromUrl request. |
| Cleanup | N/A |

MSWEBSS\_S05\_TC05\_GetObjectIdFromUrl\_Unauthenticated

|  |  |
| --- | --- |
| **S06\_OperationsOnListTemplates** | |
| Test case ID | MSWEBSS\_S06\_TC01\_GetListTemplates\_Unauthenticated |
| Description | This test case aims to verify the GetListTemplates operation when the user is unauthenticated. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetListTemplate request.   **Input parameters:**  None   1. The server will return an HTTP 401 error when unauthorized user sends GetListTemplate request. |
| Cleanup | N/A |

MSWEBSS\_S06\_TC01\_GetListTemplates\_Unauthenticated

|  |  |
| --- | --- |
| **S06\_OperationsOnListTemplates** | |
| Test case ID | MSWEBSS\_S06\_TC02\_GetListTemplates\_Succeed |
| Description | This test case is used to call GetListTemplates operation successfully. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Send a GetListTemplate request.   **Input parameters:**  None |
| Cleanup | N/A |

MSWEBSS\_S06\_TC02\_GetListTemplates\_Succeed

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC01\_UpdateColumns\_InvalidFieldDefinition |
| Description | This test case aims to verify the UpdateColumns operation with invalid updateFields, updateFields or deleteFields which specify an invalid FieldDefinition element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters have the following combinations:**   1. newFields: Specify an invalid field definition   updateFields: null  deleteFields: null   1. newFields: null   updateFields: Specify an invalid field definition  deleteFields: null   1. newFields: null   updateFields: null  deleteFields: Specify an invalid field definition   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x80004005" and Error Text not null will be returned from the server. 2. The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when an invalid Field element is passed in any of the parameters. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC01\_UpdateColumns\_InvalidFieldDefinition

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC02\_GetColumns\_WithInvalidColumn |
| Description | This test case aims to verify the GetColumns operation when site refer to service has invalid column information. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user 2. Send a **GetColumns** request.   **Input parameters**  None.   1. A SOAP exception will returned from server. 2. The result returned in step 3 is checked if the operation including ErrorCode element. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC02\_GetColumns\_WithInvalidColumn

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC03\_UpdateColumns\_AllFieldsValid |
| Description | This test case aims to verify UpdateColumns operation to add, update, or delete one or more specified existing columns on the context site and all child sites within its hierarchy. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: valid  updateFields: valid  deleteFields: valid   1. The server will return the UpdateColumnsResult. 2. The result returned in step 3 is checked the attribute Method.ID must have the same value as the Method.ID attribute that was sent to the protocol server in the UpdateColumns message of this UpdateColumnsResponse for an add, update, or delete operation. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC03\_UpdateColumns\_AllFieldsValid

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC04\_UpdateColumns\_NoMatchingName |
| Description | This test case aims to verify the UpdateColumns operation when the attribute Name of deleteFields and updateFields doesn’t included in existing columns. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters have the following combinations:**   1. newFields: null   updateFields: Specify the Name does not match to any of the already existing names.  deleteFields: null   1. newFields: null   updateFields: null  deleteFields: Specify the Name does not match to any of the already existing names.   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x82000007" and Error Text not nullwill be returned from the server. 2. The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when the specified name attribute does not match any of the existing columns. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC04\_UpdateColumns\_NoMatchingName

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC05\_GetColumns\_Unauthenticated |
| Description | This test case aims to verify the GetColumns operation for the user without authorization. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetColumns request.   **Input parameters**  None   1. The server will return an HTTP 401 error when unauthorized user sends the GetColumns request. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC05\_GetColumns\_Unauthenticated

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC06\_UpdateColumns\_Unauthenticated |
| Description | This test case aims to verify the UpdateColumns operation for the user without authorization. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send an UpdateColumns request.   **Input parameters**  newFields: valid  updateFields: valid  deleteFields: valid   1. The server will return an HTTP 401 error when unauthorized user sends the UpdateColumns request. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC06\_UpdateColumns\_Unauthenticated

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC07\_UpdateColumns\_NewFieldsWithMultipleMethodNoFields |
| Description | This test case aims to verify the UpdateColumns operation with invalid newFields which doesn’t have a root element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: Specify that request has multiple Method elements, without a Fields element defined as the root element.  updateFields: null  deleteFields: invalid   1. A SOAP fault will be returned from the server when one of the newFields elements of the UpdateColumns element has multiple Method elements without a Fields element defined as the root element. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC07\_UpdateColumns\_NewFieldsWithMultipleMethodNoFields

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC08\_UpdateColumns\_UpdateFieldWithMultipleMethodNoFields |
| Description | This test case aims to verify the UpdateColumns operation with invalid updateFields which doesn’t have a root element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: null  updateFields: Specify that request has multiple Method elements, without a Fields element defined as the root element.  deleteFields: null   1. A SOAP fault will be returned from the server when one of the updateFields elements of the UpdateColumns element has multiple Method elements without a Fields element defined as the root element. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC08\_UpdateColumns\_updateFieldWithMultipleMethodNoFields

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC09\_UpdateColumns\_DeleteFieldsWithMultipleMethodNoFields |
| Description | This test case aims to verify the UpdateColumns operation with invalid DeleteFields which doesn’t have a root element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: null  updateFields: null  deleteFields: Specify that request has multiple Method elements, without a Fields element defined as the root element.   1. A SOAP fault will be returned from the server when one of the deleteFields elements of the UpdateColumns element has multiple Method elements without a Fields element defined as the root element. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC09\_UpdateColumns\_deleteFieldsWithMultipleMethodNoFields

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC10\_UpdateColumns\_ExistingColumn |
| Description | This test case aims to verify the UpdateColumns operation with invalid newFields which specifies an already existing column. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: Specify an already existing column which is wanted to be added to the site.  updateFields: null  deleteFields: null   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x80004005" and Error Text not null will be returned from the server. 2. The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when an attempt is made to add an already existing column to the site. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC10\_UpdateColumns\_ExistingColumn

|  |  |
| --- | --- |
| S07\_OperationsOnColumns | |
| Test case ID | MSWEBSS\_S07\_TC11\_UpdateColumns\_NonexistentColumn |
| Description | This test case aims to verify the UpdateColumns operation with invalid deleteFields which specifies an already existing column. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: null  updateFields: null  deleteFields: Specify that a non-existing column is wanted to be deleted from the site.   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x80004005" and Error Text not null will be returned from the server.   The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when an attempt is made to delete a non-existing column to the site. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC11\_UpdateColumns\_NonexistentColumn

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC12\_UpdateColumns\_InvalidGUID |
| Description | This test case aims to verify the UpdateColumns operation with invalid GUID in deleteFields and updateFields. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: null  updateFields: Specify that an invalid GUID passed  deleteFields: Specify that an invalid GUID passed   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x80004005" and Error Text not null will be returned from the server 2. The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when an invalid GUID is passed in as the ID attribute for updateFields and deleteFields. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC12\_UpdateColumns\_InvalidGUID

|  |  |
| --- | --- |
| **S07\_OperationsOnColumns** | |
| Test case ID | MSWEBSS\_S07\_TC13\_UpdateColumns\_NoNameOrDisplayName |
| Description | This test case aims to verify the UpdateColumns operation when the attributes name or display name are not included in newFields. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send an UpdateColumns request.   **Input parameters:**  newFields: without Name or DisplayName  updateFields: null  deleteFields: null   1. A UpdateColumnsResponseUpdateColumnsResult object with error code "0x80004005" and Error Text not null will be returned from the server 2. The result object returned in step 3 is checked for the error code 0x80004005 and ErrorText will be returned when neither the Name nor the DisplayName attribute is passed in a Field element of newFields. |
| Cleanup | N/A |

MSWEBSS\_S07\_TC13\_UpdateColumns\_NoNameOrDisplayName

|  |  |
| --- | --- |
| **S08\_OperationsOnCSS** | |
| Test case ID | MSWEBSS\_S08\_TC01\_CustomizeCss\_Unauthenticated |
| Description | This test case aims to verify the CustomizeCss operation with a valid cssFile without authorization. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: valid   1. The server will return an HTTP 401 error when unauthorized user sends the CustomizeCss request. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC01\_CustomizeCss\_Unauthenticated

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC02\_RevertCss\_CssFileNotExist |
| Description | This test case aims to verify the RevertCss operation when cssFile doesn’t exist on the protocol server. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertCss request.   **Input parameters**  cssFile: Specifies a cssFile doesn’t exist on the server.   1. A SOAP fault with error code 0x80070002 which specified the cssFile doesn’t exist on the server wiil be returned from server. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC02\_RevertCss\_CssFileNotExist

|  |  |
| --- | --- |
| **S08\_OperationsOnCSS** | |
| Test case ID | MSWEBSS\_S08\_TC03\_CustomizeCss\_CssFileBlankOrMissing |
| Description | This test case aims to verify the CustomizeCss operation when the cssFile element is blank or missing. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: specify a blank cssFile parameter.   1. A SOAP fault contains error code 0x82000001 will be returned from the server. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: null   1. A SOAP fault contains error code 0x82000001 will be returned from the server. 2. The result object returned in step 6 is checked for the meaning of Error Code 0x82000001 is Blank cssFile specified, or cssFile element missing. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC03\_CustomizeCss\_CssFileBlankOrMissing

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC04\_CustomizeCss\_NoFileExtension |
| Description | This test case aims to verify the CustomizeCss operation when the cssFile element has no file name extension. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: Specifies a cssFile which has no file name extension.   1. A SOAP fault contains error code 0x80131600 which indicated the cssFile has no file extension will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC04\_CustomizeCss\_NoFileExtension

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC05\_CustomizeCss\_CssFileNotExist |
| Description | This test case aims to verify the CustomizeCss operation when the cssFile does not exist on the protocol server. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: Specifies a cssFile doesn’t exist on the server.   1. A SOAP fault contains error code 0x80070002 which indicated the cssFile does not exist on the protocol server will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC05\_CustomizeCss\_CssFileNotExist

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC06\_CustomizeCss\_ContainAsterisk |
| Description | This test case aims to verify the CustomizeCss operation when the cssFile contains an asterisk. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: contains an asterisk   1. A SOAP fault contains error code 0x80131600 which means client secified a cssFile value containing an asterisk will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC06\_CustomizeCss\_ContainAsterisk

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC07\_RevertCss\_Unauthenticated |
| Description | This test case aims to verify the RevertCss operation when the user is not authorized. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a RevertCss request.   **Input parameters**  cssFile: valid   1. The server will return an HTTP 401 error when unauthorized user sends RevertCss request. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC07\_RevertCss\_Unauthenticated

|  |  |
| --- | --- |
| **S08\_OperationsOnCSS** | |
| Test case ID | MSWEBSS\_S08\_TC08\_RevertCss\_Succeed |
| Description | This test case aims to verify RevertCss operation to let the customizations of the context site defined by the given CSS file return to the default style. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a RevertCss request.   **Input parameters**  cssFile: valid   1. The server will respond by changing the configuration of the site to use the default version of the specified CSS file when the transport protocol is HTTP, then having received the response successfully. |
| Cleanup | N/A |

MSWEBSS\_S08\_TC08\_RevertCss\_Succeed

|  |  |
| --- | --- |
| **S08\_OperationsOnCSS** | |
| **Test case ID** | MSWEBSS\_S08\_TC09\_RevertCss\_CssFileBlankOrNull |
| **Description** | This test case aims to verify the RevertCss operation when cssFile is blank or null. |
| **Prerequisites** | Common prerequisites |
| **Test execution steps** | 1. Common steps of authorized user. 2. Send a RevertCss request.   **Input parameters have following combinations:**   1. cssFile: Specifies a blank cssFile parameter. 2. cssFile: Specifies a null cssFile parameter. 3. A SOAP fault with error code 0x82000001 which indicated the cssFile is blank or null will be returned from server. 4. A SOAP fault with error code 0x82000001 which indicated the cssFile is blank or null will be returned from server. |
| **Cleanup** | N/A |

MSWEBSS\_S08\_TC09\_RevertCss\_CssFileBlankOrNull

|  |  |
| --- | --- |
| S08\_OperationsOnCSS | |
| Test case ID | MSWEBSS\_S08\_TC10\_CustomizeCssFileMatchName |
| Description | This test case aims to verify the RevertCss operation when cssFile does not exist on server. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a CustomizeCss request.   **Input parameters**  cssFile: Specifies a cssFile with empty name, Like String.empty.   1. Send a CustomizeCss request.   **Input parameters**  cssFile: Specifies a cssFile have no .css extension. Like "cus.cxx" |
| Cleanup | N/A |

MSWEBSS\_S08\_TC10\_ CustomizeCssFileMatchName

|  |  |
| --- | --- |
| **S09\_OperationsOnWeb** | |
| **Test case ID** | MSWEBSS\_S09\_TC01\_GetWeb\_Unauthenticated |
| **Description** | This test case aims to verify the GetWeb operation when the user is not authorized. |
| **Prerequisites** | Common prerequisites |
| **Test execution steps** | 1. Common steps of unauthorized user. 2. Send a GetWeb request.   **Input parameters**  webUrl: valid   1. The server will return an HTTP 401 error when unauthorized user sends GetWeb request. |
| **Cleanup** | N/A |

MSWEBSS\_S09\_TC01\_GetWeb\_Unauthenticated

|  |  |
| --- | --- |
| **S09\_OperationsOnWeb** | |
| Test case ID | MSWEBSS\_S09\_TC02\_GetWeb\_Succeed |
| Description | This test case aims to verify GetWeb operation to get the Title, URL, Description, Language, and theme properties of the specified site. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetWeb request.   **Input parameters**  webUrl: valid   1. The server will return the title, URL, description, language and theme properties of the specified site. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC02\_GetWeb\_Succeed

|  |  |
| --- | --- |
| S09\_OperationsOnWeb | |
| Test case ID | MSWEBSS\_S09\_TC03\_GetWeb\_InvalidUrl |
| Description | This test case aims to verify the GetWeb operation when an invalid URL is passed in the site. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetWeb request.   **Input parameters**  webUrl: invalid   1. A SOAP fault with error code 0x82000001 and the Error Text not null or empty will be returned from server. 2. The result object returned in step 3 is checked for when the invalid URL passed to the site the Error Code 0x82000001 is returned. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC03\_GetWeb\_InvalidUrl

|  |  |
| --- | --- |
| S09\_OperationsOnWeb | |
| Test case ID | MSWEBSS\_S09\_TC04\_GetWeb\_NoWebUrl |
| Description | This test case aims to verify the GetWeb operation when no webUrl is supplied for. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetWeb request.   **Input parameters**  webUrl: specify a webUrl which is missing   1. A SOAP fault with error code 0x82000001 and the Error Text is not null or empty will be returned from server. 2. The result object returned in step 3 is checked when the invalid URL passed to the site the Error Code 0x82000001 is returned. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC04\_GetWeb\_NoWebUrl

|  |  |
| --- | --- |
| **S09\_OperationsOnWeb** | |
| **Test case ID** | MSWEBSS\_S09\_TC05\_GetWebCollection\_Unauthenticated |
| **Description** | This test case aims to verify the GetWebCollection operation when the user is not authorized. |
| **Prerequisites** | Common prerequisites |
| **Test execution steps** | 1. Common steps of unauthorized user. 2. Send a GetWebcollection request.   **Input parameters**  None   1. The server will return an HTTP 401 error when unauthorized use sends GetWebcollection request. |
| **Cleanup** | N/A |

MSWEBSS\_S09\_TC05\_GetWebCollection\_Unauthenticated

|  |  |
| --- | --- |
| S09\_OperationsOnWeb | |
| Test case ID | MSWEBSS\_S09\_TC06\_GetWebCollection\_Succeed |
| Description | This test case verifies the GetWebCollection operation succeeds. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user and initialize Service:   AdapterHelper.GetTransportType(),  SoapProtocolVersion.Soap11,  UserAuthentication.Authenticated,  ConstString.DefaultServerUrl   1. Send a **GetWebCollection** request.   **Input parameters**  None.   1. The server will respond with **Webs** message. If the child element including Title and URL properties in the array, |
| Cleanup | N/A |

MSWEBSS\_S09\_TC06\_GetWebCollection\_Succeed

|  |  |
| --- | --- |
| S09\_OperationsOnWeb | |
| Test case ID | MSWEBSS\_S09\_TC07\_GetAllSubWebCollection\_Unauthenticated |
| Description | This test case aims to verify the GetAllSubWebCollection operation when the user is not authorized. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetAllSubWebcollection request.   **Input parameters**  None  The server will return an HTTP 401 error when unauthorized user sends GetAllSubWebcollection request. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC07\_GetAllSubWebCollection\_Unauthenticated

|  |  |
| --- | --- |
| S09\_OperationsOnWeb | |
| Test case ID | MSWEBSS\_S09\_TC08\_GetAllSubWebCollection\_Succeed |
| Description | This test case aims to verify GetAllSubWebCollection operation when the operation is succeed. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetAllSubWebCollection request.   **Input parameters**  None   1. The server will return a list of the Titles and URLs of all sites in the site collection when the transport protocol is HTTP and the SOAP version is SOAP12. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC08\_GetAllSubWebCollection\_Succeed

|  |  |
| --- | --- |
| **S09\_OperationsOnWeb** | |
| Test case ID | MSWEBSS\_S09\_TC09\_GetWebDefinition |
| Description | This test case aims to call GetWeb operation to verify that the WebDefinition complex type that specifies a single site. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Sends a **GetWeb** request message.   **Input parameters**  title, description, language, mainUrl, theme, FarmId and Id   1. A set of XML attributes will be returned from the server. |
| Cleanup | N/A |

MSWEBSS\_S09\_TC09\_GetWebDefinition

|  |  |
| --- | --- |
| **S09\_OperationsOnWeb** | |
| Test case ID | MSWEBSS\_S09\_TC10\_GetWebResponseWithValidXml |
| Description | This test case aims to call GetWeb operation to verify that the SOAP body contains a GetWebResponse element. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a **GetWeb** request.   **Input parameters**  webUrl: Valid   1. The protocol server responds with a **GetWebResponse** response message including a **GetWebResult** element, 2. The result object returned in step 3 is checked for when the response to the request **GetWebResult** including a **Web** element and its type is **WebDefinition**, |
| Cleanup | N/A |

MSWEBSS\_S09\_TC10\_GetWebResponseWithValidXml

|  |  |
| --- | --- |
| S10\_OperationsOnActivatedFeatures | |
| Test case ID | MSWEBSS\_S10\_TC01\_GetActivatedFeaturesAboveWSS3 |
| Description | This test case verifies the optional behavior of the GetActivatedFeatures operation. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Sends a GetActivatedFeaturesSoapIn request message.   **Input parameters**  None   1. A SOAP fault will not be returned a list of activated features on the site and on the parent site collection. |
| Cleanup | N/A |

MSWEBSS\_S10\_TC01\_GetActivatedFeaturesAboveWSS3

|  |  |
| --- | --- |
| **S10\_OperationsOnActivatedFeatures** | |
| Test case ID | MSWEBSS\_S10\_TC02\_GetActivatedFeatures\_Succeed |
| Description | This test case aims to verify GetActivatedFeatures operation to get a list of activated features on the site and on the parent site collection. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Send a GetAllSubWebCollection request.   **Input parameters**  None   1. The GetActivatedFeaturesResult will be returned from the server. 2. Call GetObjectID method in SUT control adapter to get GUID of the feature on the site to check the result returned in step 3. If the GetActivatedFeaturesResponse element contains a single string, formatted as a comma-delimited list of GUIDs, each identifying an activated feature on the site. 3. Call GetObjectID method in SUT control adapter to get GUID of the feature on the parent site collection to check the result returned in step 3. If the value of GetActivatedFeaturesResult is a comma-delimited list of GUIDs of features activated on the site, followed by a comma-delimited list of GUIDs of features activated on the parent site collection. |
| Cleanup | N/A |

MSWEBSS\_S10\_TC02\_GetActivatedFeatures\_Succeed

|  |  |
| --- | --- |
| S10\_OperationsOnActivatedFeatures | |
| Test case ID | MSWEBSS\_S10\_TC03\_GetActivatedFeatures\_Unauthenticated |
| Description | This test case aims to verify the GetActivedFeatures operation when the user is not authorized. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of unauthorized user. 2. Send a GetAllSubWebcollection request.   **Input parameters**  None   1. The server will return an HTTP 401 error when unauthorized user sends GetAllSubWebcollection request. |
| Cleanup | N/A |

MSWEBSS\_S10\_TC03\_GetActivatedFeatures\_Unauthenticated

|  |  |
| --- | --- |
| S10\_OperationsOnActivatedFeatures | |
| Test case ID | MSWEBSS\_S10\_TC04\_GetActivatedFeaturesResponseWithCommaDelimitedGUID |
| Description | This test case verifies that the GetActivatedFeatures operation aims to verify get a comma-delimited list of GUIDs of activated features on the site or on the parent site collection. |
| Prerequisites | Common prerequisites |
| Test execution steps | 1. Common steps of authorized user. 2. Sends a GetActivatedFeaturesSoapIn request message.   **Input parameters**  None   1. A SOAP fault will be returned from the server when contains a single string, formatted as a comma-delimited list of **GUIDs**. |
| Cleanup | N/A |

MSWEBSS\_S10\_TC04\_GetActivatedFeaturesResponseWithCommaDelimitedGUID