

TEST CASE – H5 INVALIDATION

Version: 1.3
Author: Cina Arjmand

netcompany

Document history

| Version | Date | Author | Status | Comments |
|---------|------------|-----------------|-----------|---|
| 0.1 | 01-03-2024 | Cina Arjmand | Draft | First draft |
| 0.2 | 16-10-2024 | Emilie Madsen | Draft | Review |
| 1.0 | 24-10-2024 | Emilie Madsen | Completed | |
| 1.1 | 16-01-2025 | Michael Larsson | Completed | Updated routing for Standard folder |
| 1.2 | 24-06-2025 | Johannes Bæk | Completed | Removed references to TFE02 |
| 1.3 | 09-07-2025 | Johan Lövberg | Completed | Updated Process flow, Test scenario and XML example |

Table of contents

1 Pre-conditions..... 3

2 Process flow..... 4

3 Test Scenarios 5

 3.1 Description of test scenarios 5

 3.2 Test scenario 1 – Acceptance..... 6

 3.3 Test scenario 2 – Rejection..... 8

 3.3.1 XML example..... 9

1 Pre-conditions

To complete the functional test case for “H5 Invalidation”, the company must have an established connection to the AS4 Gateway, functioning system user to DMS Import, see [Connectivity Guide & System Guide](#).

Furthermore, ensure that you for this test case have the correct URLs, Services, and Actions, as seen below. In place of {CVR} and {UUID}, fill out your own CVR and UUID in the URLs below. While the URLs and Services are the same for all the test cases, the Actions however, will change depending on the test case you are performing, please keep an eye out that you have the correct Action for the test case in hand.

To submit the XML for this test case, use the following endpoints:

| URL | TFE Service | Action |
|---|-------------|------------------------|
| https://secureftpgatewaytest.skat.dk:6384/exchange/CVR_{CVR}_UI_{UUID} | DMS.Import2 | Declaration.Submit |
| https://secureftpgatewaytest.skat.dk:6384/exchange/CVR_{CVR}_UI_{UUID} | DMS.Import2 | Declaration.Invalidate |

For retrieving the notification, use the following endpoint:

| URL | TFE Service | Action |
|---|-------------|--------------|
| https://secureftpgatewaytest.skat.dk:6384/exchange/CVR_{CVR}_UI_{UUID} | DMS.Import2 | Notification |

Please ensure that the “**Test Case – H5 Standard**” have been completed **before** you try to complete the “**Test Case – H5 Invalidation**” scenarios.

2 Process flow

The process flow for a H5 invalidation declaration can be seen in Figure 1 below:

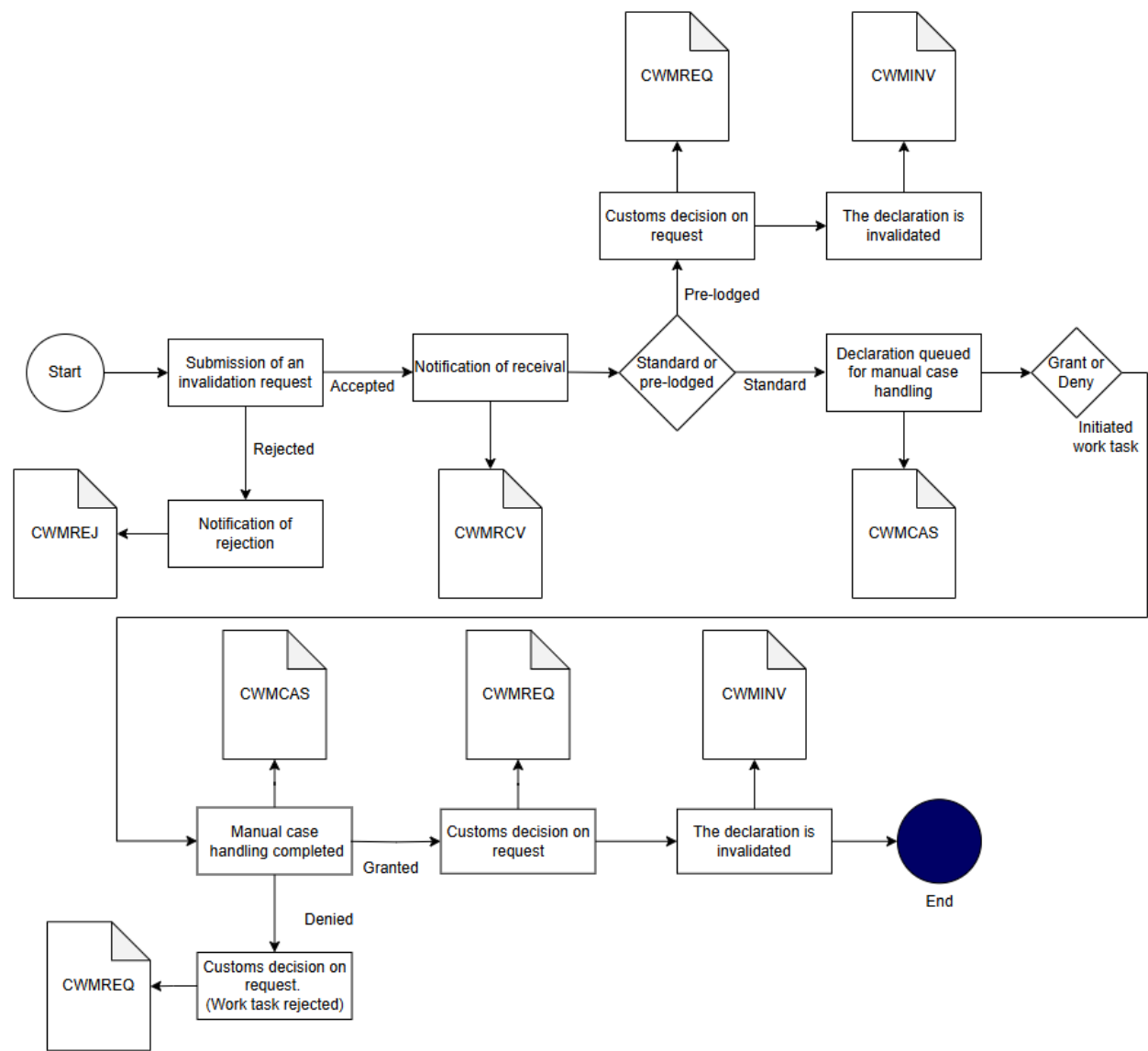


Figure 1 : Process flow for H5 invalidation

3 Test Scenarios

To complete this test case, test scenario 1 and 2 must be passed successfully. Descriptions and specifics for each test can be found in the coming sections. Be sure to go through the scenarios thoroughly. It is recommended that you use the attached files to test the notifications.

| Test no. | Test scenario | Passed/Failed |
|----------|---|--------------------------|
| 1 | Submit H5 – Invalidation, which is accepted | <input type="checkbox"/> |
| 2 | Submit H5 – Invalidation, which is rejected | <input type="checkbox"/> |

3.1 Description of test scenarios

The following sections will describe the aim of each specific scenario and provide the desired results of the test scenarios. It is worth noting that there can be changes in the notifications provided by DMS Import as the solution is still in development.

The purpose of the test scenarios is to ensure your connection to the system, and that you as a company can receive the correct notifications when sending in XMLs. Each step describes what you need to do when going through the test case. For each case there will be an XML available in the same folder where you found this test case document.

3.2 Test scenario 1 – Acceptance

The aim of this scenario is to get a notification that the H5 Invalidation XML has been accepted.

The following table shows the necessary test steps for completing this scenario as well as expected results. For the first step it is recommended using the provided declaration **XML** found in the Test Case folder. Secondly, make sure that your **LRN** and **MRN** corresponds to those of the H5 – Standard declaration, and the Submitter field is correct (this is done by putting your **CVR** in the Submitter/Name and Submitter/ID fields). You should also make sure that the Amendment field is filled with a valid **ChangeReasonCode** and a **ChangeReasonText** of at least 10 characters. The placeholders are denoted in the XML by double curly brackets {{placeholder}}.

| Step no. | Description of steps | Expected result | Passed |
|----------|---|--|--------------------------|
| 1 | Use a H5 Standard XML . It is found within the H5 Standard folder that is found in the parent directory at H5 Test Cases (remember to replace the placeholders) | You should have the test XML ready for the next step | <input type="checkbox"/> |
| 2 | Submit the declaration using the conditions found under Pre-conditions | The declaration should be sent to the system | <input type="checkbox"/> |
| 3 | Pull the notifications | You should be able to pull the notification from the system | <input type="checkbox"/> |
| 4 | Acceptance of submission of declaration. (Note that if you receive the CWMCLE notification that you won't be able to invalidate the declaration without an invalidation 4c as the declaration has been released. If you got this notification, send a new H5 standard and try again.) | Receive the CWMACC notifications | <input type="checkbox"/> |
| 5 | Use the provided "Test Case – H5 Invalidation" XML found in the test case folder (remember to replace the placeholders – {{LRN}} and {{MRN}} should match the MRN and LRN from the initially submitted H5 Standard declaration). | Find the H5 Invalidation XML in the H5 Invalidation test case folder | <input type="checkbox"/> |
| 6 | Submit the declaration using the conditions found under Pre-conditions | The declaration should be sent to the system | <input type="checkbox"/> |
| 7 | Pull the notifications | You should be able to pull notification from the system | <input type="checkbox"/> |
| 8 | Successful validation of submission of invalidation request by receiving the CWMRCV and CWMCAS notifications | You should receive the CWMRCV and CWMCAS notifications | <input type="checkbox"/> |
| 9 | After having received the CWMCAS , read the AdditionalInformation field regarding the status. This notification informs you of a manual acceptance work task being created. | You should find a StatementCode field with the number 41 (CWMCAS Queued). Then in the StatementDescription field there should be an explanation | <input type="checkbox"/> |

| | | | |
|----|--|--|--------------------------|
| 10 | <p>When the queued request is completed, receive the final CWMCAS notification with changes to AdditionalInformation. This informs you that the worktask is completed.</p> <p>A manual work task needs to be manually approved by a customs official. Until a customs official has manually processed your declaration, you won't receive any further notifications. As such, the declaration may remain on the previous step for some time if so, pull the notification again to get these notifications.</p> | One notification with StatementCode field 43 (CWMCAS Completed). | <input type="checkbox"/> |
| 11 | Acceptance of submission of invalidation request by receiving the CWMREQ and CWMINV notifications | Receive notifications CWMREQ and CWMINV to pass the test | <input type="checkbox"/> |

3.3 Test scenario 2 – Rejection

The aim of this scenario is for the EO to get a notification that the H5 Invalidation XML has been rejected.

The following table shows the necessary test steps for completing this scenario as well as expected results. For the first step it is recommended using the provided declaration **XML** found in the Test Case folder. Secondly, make sure that your **LRN** and **MRN** corresponds to those of the H5 – Standard declaration, and the Submitter field is correct (this is done by putting your **CVR** in the Submitter/Name and Submitter/ID fields). You should also make sure that the Amendment field is filled with a valid **ChangeReasonCode** and a **ChangeReasonText** of at least 10 characters. The placeholders are denoted in the XML by double curly brackets {{placeholder}}.

| Step no. | Description of steps | Expected result | Passed |
|----------|---|---|--------------------------|
| 1 | Use a H5 Standard XML . It is found within the H5 Standard folder that is found in the parent directory at H5 Test Cases (remember to replace the placeholders) | You should have the test XML ready for the next step | <input type="checkbox"/> |
| 2 | Submit the declaration using the conditions found under Pre-conditions | The declaration should be sent to the system | <input type="checkbox"/> |
| 3 | Pull the notifications | You should be able to pull the notification from the system | <input type="checkbox"/> |
| 4 | Acceptance of submission of declaration. (Note that if you receive the CWMCLE notification that you won't be able to invalidate the declaration without an invalidation 4c as the declaration has been released. If you got this notification, send a new H5 standard and try again.) | Receive the CWMACC notification | <input type="checkbox"/> |
| 5 | Use the provided "Test Case – H5 Invalidation" XML found in the test case folder (remember to replace the placeholders – {{LRN}} and {{MRN}} should match the MRN and LRN from the initially submitted H5 Standard declaration). | Find the H5 Invalidation XML in the H5 Invalidation test case folder | <input type="checkbox"/> |
| 6 | Adapt the H5 Invalidation XML with the information with the invalid data in the element Amendment as shown in section 0, to ensure rejection of the XML. | Enter element from section 0 | <input type="checkbox"/> |
| 7 | Submit the declaration using the conditions found under Pre-conditions | The declaration should be sent to the system | <input type="checkbox"/> |
| 8 | Pull the notifications | You should be able to pull notification from the system | <input type="checkbox"/> |
| 9 | Rejection of submission of invalidation request by receiving the CWMREJ notification | Receive notification CWMREJ and pass the scenario | <input type="checkbox"/> |

3.3.1 XML example

```
<ns2:Amendment>  
  <ns2:SequenceNumeric>1</ns3:SequenceNumeric>  
  <ns2:ChangeReasonCode>22</ns3:ChangeReasonCode>  
  <ns2:ChangeReasonText>Invalid</ns3:ChangeReasonText>  
</ns2:Amendment>
```