TEST CASE – VICTUALLING (PROVIANT) B1

Version: 1.1
Author: Jesper Sture Bergendorff

netcompany

Document history

Version	Date	Author	Status	Comments
1.0	03-11-2023	Jesper Sture Bergendorff	Done	First edition
1.1	02-07-2024	Johannes Fuglseth Bæk	Updated	Changed URL's in section 1 and verified flow.

Table of contents

1 Pre-conditions

To complete the functional test case for “Victualling (Proviant)”, the company must have an established connection to the AS4-gateway, and have a registered system user for DMS Export, 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 endpoint:

URL	UFE Service	TFE Service	Action
https://secureftpgatewaytest.skat.dk:6384/exchange/CVR_{CVR}_UI_{UUID}	DMS.Export2	DMS.Export	Declaration.Submit

For retrieving notifications, use the following endpoint:

URL	UFE Service	TFE Service	Action
https://secureftpgatewaytest.skat.dk:6384/exchange/CVR_{CVR}_UI_{UUID}	DMS.Export2	DMS.Export	Notification

1.1 Process flow

The process flow for a B1 standard declaration, which this Victualling (Proviant) B1 test case will also follow, can be seen in Figure 1-1 below. **NB!**: Notice that there are two rounds of manual acceptance for a B1 Victualling (Proviant) declaration, the first is the usual case handling of the declaration itself, and the second is manual acceptance of the Victualling procedure. For an even more detailed chart of the Standard B1 flow, see our B1 Standard test case in the dms-public GitHub repo.

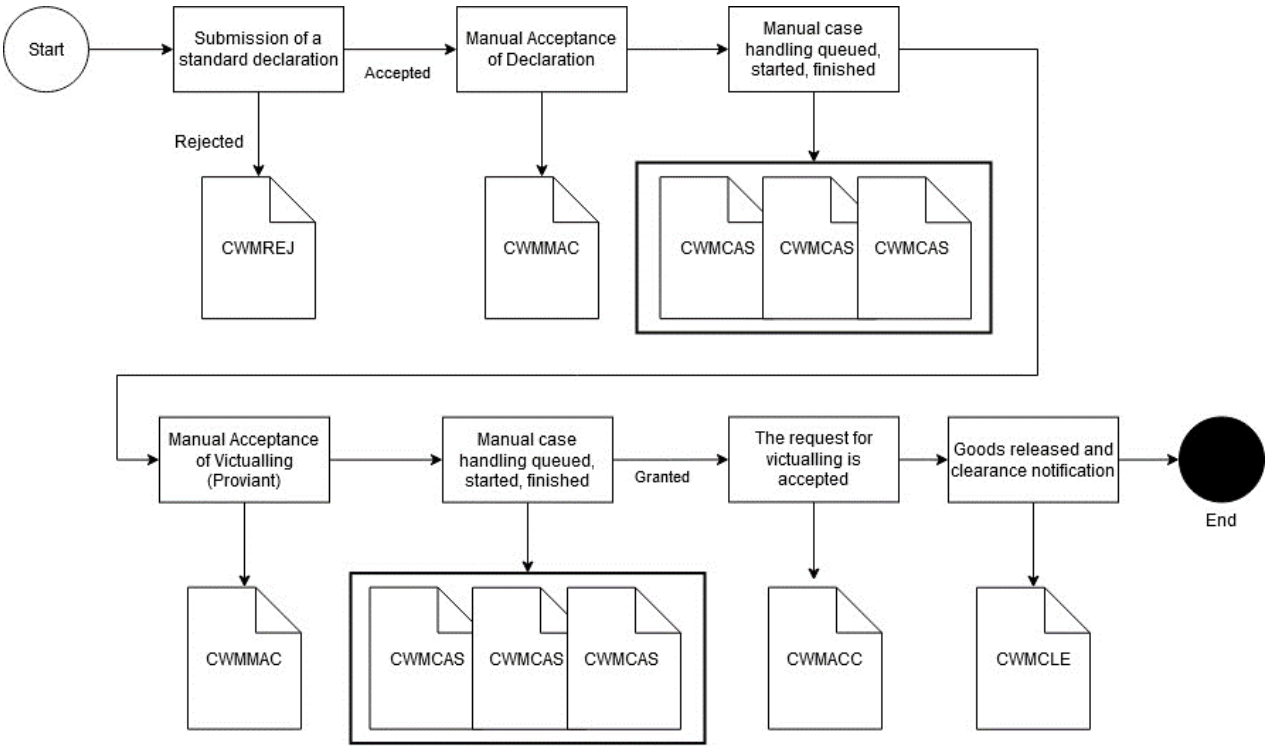

Figure 1-1 – Victualling (Proviant) export declaration flowchart

2 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 B1 – Victualling (Proviant) declaration, which is accepted	☐
2	Submit B1 – Victualling (Proviant) declaration, which is rejected	☐

2.1 Description of test scenarios

The following sections will describe the aim and desired results of each specific test scenario.

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. An XML for the Victualling (Proviant) B1 declaration is made available in the same folder where you found this test case document. This XML file may need to be changed by the user according to the test scenario.

Since this is a functional test case, it does not complete the declaration flow fully. To test a complete flow, please refer to the test case "TEST CASE – B1 WITH IE507 AND IE590" in the Test cases\Test Cases IE messages\Test Case - IE507 Arrival At Exit folder on the dms-public GitHub repo.

2.2 Test scenario 1 – Acceptance

The aim of this scenario is to get a notification that the Victualling (Proviant) B1 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** is unique, and that the **Submitter** field is correct (this is done by putting your CVR in the **Submitter/Name** and **Submitter/ID** fields).

Step no.	Description of steps	Expected result	Passed
1	Get the provided "Test Case – Victualling (Proviant) B1" XML found in the test case folder (remember to replace the **{{LRN}}** and **{{CVR}}** placeholders)	You should have the test **XML** ready for the next step	☐
2	Fill out the <EXTENSION> elements with the desired data to match your test needs. See section 2.2.1 for a further description	**GoodsLocation** (16 15 000 000) element should resemble section 2.2.1	☐
3	Submit the declaration using the conditions found under Pre-conditions	The declaration should be sent to the system	☐
4	Pull the notifications	You should be able to pull notifications from the system	☐
5	Acceptance of submission of B1 declaration by receiving the **CWMACC and CWMCLE** notifications	Receive notifications **CWMACC and CWMCLE** and pass the test	☐

2.2.1 XML example

NB!: All of the following Extensions must be declared in any victualling declaration, otherwise the declaration will be rejected. As an example, if no liquor is being exported, simply declare value as zero like so:

```
<ns4:Key>LIQ_AMOUNT</ns4:Key>
<ns4:Value>0</ns4:Value>
```

Below is the full list of Extensions, also found in the Victualling (Proviant) B1.xml file. Please note that the elements below are filled out with example values, and that you must fill them out with the appropriate values for your declaration.

```
<ns3:Extensions>
        <ns4:SequenceNumeric>1</ns4:SequenceNumeric>
        <ns4:Key>VEHC_DRIVER</ns4:Key>
        <ns4:Value>1</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>2</ns4:SequenceNumeric>
        <ns4:Key>VEHC_TYPE</ns4:Key>
        <ns4:Value>TYPE</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>3</ns4:SequenceNumeric>
        <ns4:Key>HOME_PORT</ns4:Key>
        <ns4:Value>PORT Aalborg</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>4</ns4:SequenceNumeric>
        <ns4:Key>DEST_EXP_R</ns4:Key>
        <ns4:Value>København</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>5</ns4:SequenceNumeric>
        <ns4:Key>DUR_EXP</ns4:Key>
        <ns4:Value>125</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>6</ns4:SequenceNumeric>
        <ns4:Key>PLACE_UNLO</ns4:Key>
        <ns4:Value>Aalborg</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>7</ns4:SequenceNumeric>
        <ns4:Key>NR_CREW</ns4:Key>
```

```
        <ns4:Value>20</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>8</ns4:SequenceNumeric>
        <ns4:Key>LIQ_AMOUNT</ns4:Key>
        <ns4:Value>484</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>9</ns4:SequenceNumeric>
        <ns4:Key>WINE_AMOUNT</ns4:Key>
        <ns4:Value>15</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>10</ns4:SequenceNumeric>
        <ns4:Key>CIGARETTE_U</ns4:Key>
        <ns4:Value>54</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>11</ns4:SequenceNumeric>
        <ns4:Key>CIGAR_U</ns4:Key>
        <ns4:Value>4</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>12</ns4:SequenceNumeric>
        <ns4:Key>TOBAC_KG</ns4:Key>
        <ns4:Value>7</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>13</ns4:SequenceNumeric>
        <ns4:Key>BEER_AMOUNT</ns4:Key>
        <ns4:Value>100</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>14</ns4:SequenceNumeric>
        <ns4:Key>OTHER_SUP</ns4:Key>
        <ns4:Value>0</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
    <ns3:Extensions>
        <ns4:SequenceNumeric>15</ns4:SequenceNumeric>
        <ns4:Key>DESC_DES_VIC</ns4:Key>
        <ns4:Value>86</ns4:Value>
        <ns4:DataType>text</ns4:DataType>
    </ns3:Extensions>
```

2.3 Test scenario 2 – Rejection

The aim of this scenario is for the EO to get a notification that the B1 Standard XML has been rejected.

The following table shows the test steps necessary to complete this scenario as well as expected results. For the first step it is recommended adapting the provided declaration XML found in the **Test Case** folder. Secondly, make sure that your **LRN** is unique, and that the **Submitter** field is correct (this is done by putting your CVR in the **Submitter/Name** and **Submitter/ID** fields).

Step no.	Description of steps	Expected result	Passed
1	Use the provided "Test Case – Victualling (Proviant) B1" XML found in the test case folder (remember to replace the **{{LRN}}** and **{{CVR}}** placeholders)	You should have the test **XML** ready for the next step	☐
2	To achieve a rejection, enter information from section 2.3.1 with the invalid data in the element **Exporter** (13 01 000 000) to ensure rejection	**Exporter** (13 01 000 000) element should resemble section 2.3.1	☐
3	Submit the declaration using the conditions found under Pre-conditions	The declaration should be sent to the system	☐
4	Pull the notifications	You should be able to pull notifications from the system	☐
5	Rejection of a submission of B1 declaration by receiving the **CWMREJ** notification	Receive **CWMREJ** and pass the test	☐

2.3.1 XML example

Input the wrong CVR number as exporter ID to get a simple rejection.

```
<ns3:Exporter>
        <ns3:ID>DK12345678</ns3:ID>
</ns3:Exporter>
```