Test protocol for receiving a

CareCommunication

January 2025

The test protocol relates to the following standard:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of the standard ENG** | **Name of the standard DK** | **Version** | **Type** |
| Standard: CareCommunication | Korrespondancemeddelelse | 4.0.0 | HL7 FHIR |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | | | |
| **Version** | **Initials** | **Date** | **Description** |
| 2.1.0 | KML/KRC/TMS/OVI | 31-03-2023 | First release |
| 3.0.0 | TMS/KRC | 18-01-2024 | Update of test protocol in accordance with release 3.0 of the documentation |
| 4.0.0 | KML/KRC/TMS | 28-06-2024 | Update of test protocol in accordance with release v. 4.0.0 of the documentation. Among other things corrective actions i.e. cancellation and correction are omitted, and test step for receiving formatted text in message is removed. Test step for parallel sent CareCommunication is further described. |
| 4.0.1 | TMS | 18-07-2024 | Updated naming of test examples and added link for test examples. |
| 4.0.2 | TMS/SKS/RCH | 03-12-2024 | Clarified formulation and expectations.  Added test step 3.3.1.2-3.3.1.9 to ensure correct support for displaying information in the CareCommunication.  Updated names of test examples. |
| 4.0.3 | KML/RCH/SKS | 15-01-2025 | Removed the test step in section 3.3 regarding deceased patient (former test step 3.3.1.6), because it is optional to implement in CareCommunication and not a part of the test and certification.  Added test step 3.3.1.2 to ensure that all systems must support and display a linebreak in the free text box of the message. |
| 4.1.0 | KML/RCH/SKS | 09-04-2025 | **Section 3.2**  Updated test files in R1 and R2  Added ‘author contact’ in test steps where SUT must display attachment information.  Removed test step “*Describe how the SUT notifies the user that a CareCommunication has been received*.” In R2 and R3, because SUT shows this in R1.  Combined two test steps in R1, R2 and R3 regarding opening a received message and displaying the message information for the user.  Added test step 3.2.3.4 in R3 to ensure that SUT connects (and displays for the user) sender information to correct message segment in the message thread.  **Section 3.3**  Updated test files  Reuse of test files to ensure a reasonable flow in test steps  Moved technical test step regarding XML and JSON from (former test step) 3.3.3 to 3.3.1.  Added more explanation to technical test step 3.3.3 regarding parallel messaging.  Shortened test step 3.3.4. |

Content

[1 Introduction 4](#_Toc195082086)

[1.1 Purpose 4](#_Toc195082087)

[1.2 Prerequisites for live test 4](#_Toc195082088)

[1.3 Documentation of self-test 5](#_Toc195082089)

[1.4 Background material 6](#_Toc195082090)

[1.5 Test examples and test persons 7](#_Toc195082091)

[1.6 Test tools 7](#_Toc195082092)

[1.7 Test result 8](#_Toc195082093)

[2 Information about vendor, system under test (SUT) and test result information 9](#_Toc195082094)

[2.1 Information about the vendor 9](#_Toc195082095)

[2.2 Information about the system under test (SUT) 9](#_Toc195082096)

[2.3 Information about test results 9](#_Toc195082097)

[3 The test 10](#_Toc195082098)

[3.1. Documentation of the test 10](#_Toc195082099)

[3.2. Test of requirements to content and flow/workflows 11](#_Toc195082100)

[3.3. Test of general technical requirements 18](#_Toc195082101)

# Introduction

This is a test protocol for receiving a CareCommunication (DK: Korrespondancemeddelelse).

All documentation concerning CareCommunication and Governance (see [Background material](#_Baggrundsmaterialer)) will be the topic of testing, and the test protocol will be continuously updated to reflect the requirements in the best way possible.

Versioning of the test protocol will follow the major and minor versions of the standard but may have a patch version that is different from the standard’s patch version.

The test protocol will be available both in Danish and English. In case of any discrepancies between the two versions, the Danish version applies.

**As regards sending Acknowledgements**: To be approved, the system under test (SUT) must be approved for receiving the FHIR Acknowledgement (DK: Kvittering). This is described in the [test protocol for sending Acknowledgements](https://medcomdk.github.io/dk-medcom-acknowledgement/#2-test-and-certification).

## Purpose

The test protocol forms the basis for the tests, which must ensure that SUT complies with the established rules and requirements for the standard. The test protocol also forms the basis for the self-test that vendors carry out prior to a live test.

## Prerequisites for live test

The following prerequisites must be met prior to the live test:

1. The vendor has read the following standard documentation, including:
   * [Clinical guidelines](#_Baggrundsmaterialer_1)
   * [Use cases](#_Baggrundsmaterialer_1)
   * [Implementation Guide](#_Baggrundsmaterialer_1)
   * [Governance](#_Baggrundsmaterialer_1)
   * And other relevant materials, cf. [background material](#_Baggrundsmaterialer_1).
2. The vendor has performed self-test, approved by MedCom
3. The vendor has created relevant test persons in the system under test (SUT)
4. The vendor is using the same version of SUT during self-test and live test
5. Approval requires that the SUT is approved for sending a FHIR Acknowledgement (DK: Kvittering).

## 

## Documentation of self-test

**Self-test**

**Prior to the test, the vendor must have performed self-test, which is approved by MedCom.**

The self-test is documented by the vendor completing this test protocol.

For self-tests, only the following column must be completed by the vendor:

* [Test data]: is filled in with the file name(s) which are uploaded and downloaded
* [Actual result]: is filled in with the results of the self-test and relevant descriptions.

Other columns are reserved for MedCom.

**During the self-test the vendor must document the test results by saving relevant files and screen dumps, and subsequently send these in a combined ZIP file (together with the completed test protocol) to** [**fhir@medcom.dk**](mailto:fhir@medcom.dk)**.**

All files and screen dumps must be named with:

* Standard name
* The number of the relevant test step
* Consecutive letter
* Whether SUT is sender (S) or recipient (R) of the standard
* File type

*Example: CareCommunication \_3.4\_A\_R.xml*

## Background material

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Version** | **Link/reference** | **Description** |
| CareCommunication – documentation site |  | <https://medcomdk.github.io/dk-medcom-carecommunication/> | Documentation site with references to all relevant documentation, including:   * Clinical guidelines for application (Sundhedsfaglige retningslinjer for anvendelse) * Use cases * Technical specifications |
| Implementation Guide |  | <https://medcomfhir.dk/ig/carecommunication/> | The FHIR technical guidelines for the standard. |
| Governance for MedCom FHIR |  | <https://medcomdk.github.io/MedCom-FHIR-Communication/> | Governance for MedCom’s FHIR standards, which describes general rules for all MedCom standards and specific rules for this standard, as well as for sending. |
| SOP for MedCom’s test and certification |  | <http://svn.medcom.dk/svn/qms/Offentlig/SOPer/SOP-7.2-MedComs%20test%20og%20certificering_godkendelse.docx> | Description of test and certification of MedCom standards and other tests courses. |
| Conversion between formats |  | <https://medcomdk.github.io/dk-medcom-carecommunication/#3-conversion-service> | Is under clarification but is planned in the context of the VANS cooperation. The plan is to ensure conversion from FHIR to OIOXML and OIOXML to FHIR. Attached files and Acknowledgements are also handled via the conversion service. |

## Test examples and test persons

|  |  |  |
| --- | --- | --- |
| **Name** | **Link/reference** | **Description** |
| Test examples | <https://medcomfhir.dk/ig/carecommunicationtestscripts/testexamples.html> | Test examples used during the test and certification. |
| Overview of test persons | <https://www.medcom.dk/opslag/koder-tabeller-ydere/tabeller/nationale-test-cpr-numre> | Overview of national test CPR numbers which can be used during the test.  **Please notice**: During the test, the vendor must be able to use any of the test persons on the list. |

## Test tools

|  |  |  |
| --- | --- | --- |
| **Name** | **Link/reference** | **Description** |
| FHIR server with MedCom profiles | <https://fhir.medcom.dk/fhir/swagger-ui/> | Public server that validates against MedCom's FHIR profiles. It is permitted to use the server for testing the upload/download of FHIR resources.  Vendors can get access by inquiry this at [fhir@medcom.dk](mailto:fhir@medcom.dk). |

## Test result

The result for each test step is categorised based on the table below:

| **Marking** | **F1** | **F2** | **F3** | **F4** | **OK** | **Not relevant** |
| --- | --- | --- | --- | --- | --- | --- |
| **Evaluation** | **Critical** | **Serious** | **Significant** | **Less significant** | **Approved** | **Not an error** |

To get the test and certification approved, the test protocol must consist exclusively of [F4] as well as [OK] results. Thus, all [F1], [F2] and [F3] must, be fixed prior to final approval.

When a test step isn’t relevant for the test course, it is noted with ‘Not relevant’.

Approval requires that SUT is approved for sending a FHIR acknowledgement (DK: Kvittering). For further information, please read: [MedCom’s test and](#TestCertificering) certification.

# Information about vendor, system under test (SUT) and test result information

## Information about the vendor

This table must be completed by **the vendor** prior to the test.

|  |  |
| --- | --- |
| Company | Completed by vendor |
| Address | Completed by vendor |
| Contact person | Completed by vendor |
| Telephone | Completed by vendor |
| E-mail | Completed by vendor |

## Information about the system under test (SUT)

This table must be completed by **the vendor** prior to the test.

|  |  |
| --- | --- |
| System | Completed by vendor |
| Version | Completed by vendor |
| Description | Completed by vendor |
| Test type | Self-test  Final test/certification |

## Information about test results

This table must be completed by MedCom when the test has been completed.

|  |  |
| --- | --- |
| Test date | 2023-01-01 |
| Test location | Completed by MedCom |
| Approved | Yes  No |
| Remarks | Completed by MedCom |
| Carried out by | Completed by MedCom. The name of the MedCom responsible (initials) for this test is inserted |

# The test

This section describes the requirements which SUT must meet before final approval.

The test is divided into three sections:

1. Test of TouchStone test scripts
2. Test of requirements for content and flow/workflows
3. Test of general technical requirements

Test participants will be asked to complete tests as described in the tables.

## Test of requirements to content and flow/workflows

The purpose of these tests is to ensure that the standard is implemented with satisfactory quality, i.e. that implementation meets the business requirements for flow and content as described in the clinical guidelines and the [use case material](#_Baggrundsmaterialer). These test steps are mainly for the user interface.

The table below lists the use cases which are tested in relation to content and flow/workflows. The table also shows the direct references to the use cases in the [use case material](#_Baggrundsmaterialer).

|  |  |  |
| --- | --- | --- |
| [**Use case**](#_Baggrundsmaterialer) | **Description** | **Section** |
| R1 | Receive a CareCommunication | 3.2.1 |
| R2 | Receive a reply to a CareCommunication | 3.2.2 |
| R3 | Receive a forwarded CareCommunication | 3.2.3 |

Table 1: Table listing the use cases which must be tested

### R1: Receive a CareCommunication

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load the CareCommunication test example. | R-A-new-b10f940e-2000-4ec8-9e0b-a3fb60fc5bb5 | CareCommunication test example is loaded. |  | Choose |
|  | Describe how the SUT notifies the user that a CareCommunication has been received. |  | User is notified that a new CareCommunication has been received. |  | Choose |
|  | Demonstrate that the SUT clearly indicates that the message is a new message.  *It is recommended that the SUT clearly indicates whether the message is new, a reply or a forwarded message.* |  | It is clearly indicated in the user interface that the message is a new message. |  | Choose |
|  | Demonstrate that the user opens the received message and that SUT shows relevant information to the user. |  | The user can open the message and see the following information.   * patient id and name, * category, * message segment including message text and signature, * sender, * receiver   If included:   * specific sender * specific recipient * topic * priority * attachments |  | Choose |
|  | Demonstrate that SUT shows the attached files including title of the files, author, author contact, and time of creation of each attached file. |  | The user can find all the information regarding the attached files in SUT. |  | Choose |
|  | Demonstrate that SUT can open or download the attached files. |  | The user can open or download the attached file. |  | Choose |
|  | Demonstrate that the SUT can load and show the attached file. |  | The user can open and see the attached file. |  | Choose |

### R2: Receive a reply to a CareCommunication

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load the test example for receiving a reply to a CareCommunication. | R-B-reply-91b4b79a-520f-48cb-bb9a-9c3e27301968 | CareCommunication test example is loaded. |  | Choose |
|  | Demonstrate that the SUT clearly indicates to the user that a reply has been received.  *It is recommended that the SUT clearly indicates to the user that the message is a new reply.* |  | It is clearly indicated in the user interface that a new reply has been received. |  | Choose |
|  | Demonstrate that the user opens the received message and that SUT shows relevant information for the user. |  | The user can open the message and see the following information.   * patient id and name, * category, * new message segment including message text and signature, * previously send message segment including message text and signature, * sender, * receiver   If included:   * specific sender * specific recipient * topic * priority * attachments |  | Choose |
|  | Demonstrate that SUT shows the attached files including title of the files, author, author contact, and time of creation of each attached file. |  | The user can find all the information regarding the attached files in SUT. |  | Choose |
|  | Demonstrate that the SUT can load and show the attached file. |  | The user can open and see the attached file. |  | Choose |

### R3: Receive a forwarded CareCommunication

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load the test example for receiving a forwarded CareCommunication. | R-C-forward-8352b67a-23b2-44c5-b43a-725270a90722 | CareCommunication test example is loaded. |  | Choose |
|  | Demonstrate that the SUT clearly indicates to the user that the message is a forwarded message.  *It is recommended that the SUT clearly indicates to the user that the message is a forwarded message.* |  | It is clearly indicated in the user interface that a forwarded message has been received. |  | Choose |
|  | Demonstrate that the user opens the received message and that SUT shows relevant information to the user. |  | The user can open the message and see the following information:   * patient id and name, * category, * new message segment including message text and signature, * previously send message segment including message text and signature, * sender, * receiver   If included:   * specific sender * specific recipient * topic * priority * attachments |  | Choose |
|  | Demonstrate that it is clear to the user, which sender (signature) and sender organization that matches each individual message segment in the message thread. |  | The user can differentiate the message segments and corresponding sender and sender organization. |  | Choose |

## Test of general technical requirements

The purpose of these test steps is to ensure that the technical sending of a CareCommunication is implemented with satisfactory quality, i.e., that it meets the governance for message communication on a general level as well as governance for CareCommunication as described in 1.4.

### Display content of a CareCommunication

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **Priority:**  Load the test data and demonstrate that SUT displays priority, when it is included in the message. | R-Tek-A-0c8dddf2-c882-4b00-8870-5035279e7d01 | Priority is displayed for the user. |  | Choose |
|  | **Linebreak:**  Load the test data and demonstrate that SUT displays linebreaks in the free text box of the message (Communication.payload:string.content). | Use test data from 3.3.1.1 | Linebreaks, marked with \n (in FHIR JSON) or &#xA; (in FHIR XML), are displayed for the user in the free text box of the message. |  | Choose |
|  | **Attachments:**  Load the test data and demonstrate that SUT displays all attachments including the attachment with the URL.  Demonstrate that SUT displays all elements included in the attachment for the user (author, author contact, creation date, title, URL) | Use test data from 3.3.1.1 | All attachments, associated structured information, and the link is displayed and can be accessed by the user. |  | Choose |
|  | **Timezone:**  Load the test data and demonstrate that SUT display correct timestamps when no timezone (zulutime) is included. | Use test data from 3.3.1.1 | The displayed timestamps are adjusted to the actual date and time of the activities in the message. |  | Choose |
|  | **EpisodeOfCare-identifier:**  *If relevant for the user, display the included episodeOfCare-identifier*.  Load the test data and demonstrate that episodeOfCare-identifier is displayed correctly. | Use test data from 3.3.1.1 | The episodeOfCare-identifier is displayed correctly. |  | Choose |
|  | **Timezone:**  Load the test data and demonstrate that SUT display correct timestamps when a timezone is included. | R-Tek-B-82875f08-4b9e-4a23-b81a-e23ec1251334 | The displayed timestamps are adjusted to the actual date and time of the activities in the message. |  | Choose |
|  | **Replacement CPR:**  Load the test data and demonstrate that SUT displays the CareCommunication for a patient with a replacement-CPR-number.  Please notice, that a replacement-CPR-number can have different structures, as described [here](https://medcomfhir.dk/ig/core/StructureDefinition-medcom-core-patient#patient-identifiers). | Use test data from 3.3.1.6 | The CareCommunication is displayed for a patient with a replacement-CPR-number. |  | Choose |
|  | **Instance IDs:**  Load the test data and demonstrate that SUT displays the message, even though the instances in the message doesn’t have UUIDs. | Use test data from 3.3.1.6 | The CareCommunication is loaded and display correctly. |  | Choose |
|  | **More information**:  Load the test data and demonstrate that SUT can handle that more information than specified in the CareCommunication standard is included. | R-Tek-C-5d29cc7d-dcf7-4b50-81b1-a8d971973ff6 | The CareCommunicaion is loaded and displayed correctly. SUT ignores the added information. |  | Choose |
|  | Describe how FHIR messages are loaded into the SUT. For example, how is data loaded into the SUT – via mapping to internal format or to own FHIR infrastructure? |  | For example: FHIR messages are used directly or processed. |  | Choose |
|  | Explain how the SUT supports parsing of XML and JSON |  | For example: SUT displays the CareCommunication received in XML and JSON format. |  | Choose |
|  | Explain how the SUT validates incoming CareCommunication messages |  | It is explained how the SUT validates incoming CareCommunication messages. |  | Choose |

### Display organization name for sender and receiver

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | *In the CareCommunication message it is not required to include the Organization.name for the sender and receiver. Therefore, SUT must use SOR to get this information.*  Load the test data without Organization.name on sender and receiver and show that SUT get the names from SOR. | R-Tek-D-68f5c0f8-e33e-4302-89b0-cdea52801ec9 | SUT uses SOR to get the sender and receiver name based on the SOR-id. SUT displays correct name for the sender and receiver. |  | Choose |

### Parallel sent CareCommunications

The following test step concerns correct handling of parallel sent replies. Parallel means that two corresponding parties replies to the same CareCommunication more or less synchronous and/or due to delays in the systems or on the VANS network the CareCommunications.

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | ***Parallel sent replies:*** *to test that the SUT can handle (rare) situations where two different messages points to (is a reply to) the same message.*  1: Load *CareCommunication\_ Ex\_R-Tek-E-new-parallell*  2:Send a reply  3: Load *CareCommunication\_ Ex\_R-Tek-F-new-parallel* | R-Tek-E-new-parallel-c1edf6f3-2554-4b72-b58b-0f030ee555ba  R-Tek-F-new-parallel-0e2864ad-a6a1-4e84-a9dc-aeacee63f981 | SUT has loaded the test data and sent a reply. |  | Choose |
|  | SUT must display the messages in one message thread as they have the same communication identifier.  SUT must clearly display the correlation between replies in the user interface, so the user can see which CareCommunication the messages refer to. |  | SUT creates one message threads, including both replies.  SUT demonstrates how a correlation between replies is clearly displayed for the user. |  | Choose |
|  | Demonstrate that the SUT displays the messages in correct order, based on the timestamp in message segments with the message text or alternatively the initiator of the communication must appear as the first, followed by the message send by the replier, when loading the parallel messages. |  | The user sees the replies in correct order in the user interface. |  | Choose |
|  | Demonstrate that the user can continue the communication in the message thread by sending or receiving a new reply to the most recently received message. |  | The user can continue the communication in the thread. |  | Choose |

### Receive CareCommunication and send FHIR Acknowledgement

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data. | R-Tek-G-20fcbedb-ced8-4949-9edf-a145e0bb9b09 | The data is loaded. |  | Choose |
|  | **Acknowledgement**  Demonstrate that the SUT sends a FHIR Acknowledgement in return (DK: Kvittering). |  | SUT returns a FHIR Acknowledgement (DK: Kvittering) |  | Choose |

### Loading of FHIR messages where the received messages are not in the same order as they were sent

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **A reply is received before the new message.**  Load test data, which is a reply to a new CareCommunication, and demonstrate that the message appears as replied in the SUT. SUT makes it visible for the user that an unread message has been received.  *A CareCommunication does not arrive, and a message with an unknown message segment and communication identifier is included in a received CareCommunication.* | R-Tek-H-3e047a9d-c86a-4828-b686-6a80403df5eb | The new and replied message appears as replied in the user interface ordered by time stamp from the message segment in the same message thread as the reply. |  | Choose |
|  | Load test data for a new CareCommunication and demonstrate that the message appears as replied in the SUT.  *A new CareCommunication arrives after the reply and is included in the message thread with the same communication identifier.* | R-Tek-I-reply-order-7c5a0cc0-f0f8-4a2e-9de9-d9293598bf94 | The new and replied message appears as replied in the user interface ordered by time stamp from the message segment in the same message thread as the reply. |  | Choose |

### Handling of doublets

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **Positive doublet – identical Bundle.id:**  Load test data and demonstrate that the CareCommunication is loaded and accessible to the SUT user. | R-Tek-J-dublicate-7283eae8-0427-4bb0-8486-06b4febd416c | The SUT user can see that a CareCommunication has been received. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR Acknowledgement (Kvittering). |  | The SUT has acknowledged positively for the CareCommunication message and sent a FHIR Acknowledgement (Kvittering) of type OK (AA) to the correct receiver. |  | Choose |
|  | Load test data from 3.3.6.1 again and demonstrate that the CareCommunication, which is a duplet, is ignored and that the SUT user still only sees one CareCommunication message. |  | The SUT user still only sees that one CareCommunication message has been received. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR Acknowledgement (Kvittering) for the duplet. |  | The SUT has acknowledged positively for the doublet and sent a FHIR Acknowledgement (Kvittering) of type OK (AA) to the correct receiver. |  | Choose |
|  | **Positive doublet – different Bundle.id:**  Load test data and demonstrate that the CareCommunication is loaded and accessible to the SUT user. | R-Tek-K-dublicate-5de12884-eae7-4fd3-ba4a-d8b56be4baff | The SUT user can see that a CareCommunication has been received. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR Acknowledgement (Kvittering). |  | The SUT has acknowledged positively for the CareCommunication message and sent a FHIR Acknowledgement (Kvittering) of type OK (AA) to the correct receiver. |  | Choose |
|  | Load test data again and demonstrate that the CareCommunication, which is a duplet, is ignored and that the SUT user still only sees one CareCommunication message. | R-Tek-L-dublicate-fa97e431-332f-4605-ad37-c259dc2b8637 | The SUT user still only sees that one CareCommunication message has been received. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR Acknowledgement (Kvittering) for the duplet. |  | The SUT has acknowledged positively for the doublet and sent a FHIR Acknowledgement (Kvittering) of type OK (AA) to the correct receiver. |  | Choose |
|  | **Negative doublet:**  Load test data and demonstrate that the CareCommunication is loaded. | R-Tek-M-dublicate-inv-260a3cc1-38eb-454a-a57b-9e32eabb54a3 | The CareCommunication has been loaded. |  | Choose |
|  | Demonstrate that the SUT has sent a negative FHIR Acknowledgement (Kvittering). |  | The SUT has acknowledged negatively for the CareCommunication message and sent a FHIR Acknowledgement (Kvittering) of type fatal error (AE) to the correct receiver. |  | Choose |
|  | Load test data again and demonstrate that the CareCommunication, which is a doublet, is loaded. | R-Tek-N-dublicate-inv-260f76f2-d986-4ba1-8049-2369b1b164ce | The CareCommunication has been loaded. |  | Choose |
|  | Demonstrate that the SUT has sent a negative FHIR Acknowledgement (Kvittering) for the duplet. |  | The SUT has acknowledged negatively for the doublet and sent a FHIR Acknowledgement (Kvittering) of type fatal error (AE) to the correct receiver. |  | Choose |

### Handling of invalid messages

| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data and demonstrate how the SUT handles invalid messages, e.g. missing valid “Communcation.category”. | R-Tek-O-invalid-3d1149cd-d3e3-421a-ae31-4860decb4272 | The SUT returns a FHIR Acknowledgement (Kvittering) of type fatal error (AE), where the error is described. |  | Choose |
|  | SUT does not display the invalid CareCommunication. |  | The SUT does not show the invalid message. |  | Choose |