Test protocol for receiving a

HospitalNotification message

14-12-2023

The test protocol relates to the following standard:

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard’s name ENG** | **Standard’s Name DK** | **Version** | **Type** |
| Standard: HospitalNotification | Advis om sygehusophold | 3.0.X | HL7 FHIR |

|  |  |  |  |
| --- | --- | --- | --- |
| **Versioning** | | | |
| **Version** | **Initials** | **Date** | **Description** |
| 3.0.0 | MBK/TMS/MBU | 06-01-2023 | The content of the original test protocol has been transferred to the new template for testing MedCom's FHIR standards. In addition to clarifications, minor error corrections and layout adjustments, corrections and clarifications have been made in information and test steps regarding receipts (Acknowledgements), as well as clarifications regarding correct use of time stamps. |
| 3.0.1 | MBK/TMS | 26-01-2023 | Correction of time stamps reference as well as general clarification regarding Loading series of FHIR messages in an admission flow, where receipt of messages does not follow the order in which the messages have been generated (sent/received). |
| 3.0.2 | MBK/TMS | 01-05-2023 | Addition of test step that tests correct handling of cancellation of a message that has been corrected prior to the cancellation (test step 3.3.11.3).  Addition of a note on use of SOR (test step 3.3.12.2)  Clarification of test steps regarding use of EpisodeOfCareIdentificer (section 3.4.2)  Clarification of test step regarding handling a duplicate (section 3.4.4) |
| 3.0.3 | MBK/TMS | August 2023 | Notification statuses have been translated back to Danish, as statuses must be displayed for the user in Danish.  Naming of test files is updated in accordance with MedComs standard operating procedures. |
| 3.0.4 | TMS/MBU | September 2023 | Loosened up the requirements for executing TouchStone test scripts for receiving HospitalNotifications (test step 3.2.1.1).  Included requirements for executing TouchStone test scripts for receiving HospitalNotifications and sending an Acknowledgement (test step 3.4.1.2). |
| 3.0.5 | MBU | 14-12-23 | Test step 3.4.7 Handling of HospitalNotifcation in XML and JSON formats is added |

Table of Contents

[1 Introduction 4](#_Toc146700418)

[1.1 Purpose 5](#_Toc146700419)

[1.2 Prerequisites for live test 5](#_Toc146700420)

[1.3 Documentation of self-test 6](#_Toc146700421)

[1.4 Background materials 7](#_Toc146700422)

[1.5 Test examples and test persons 7](#_Toc146700423)

[1.6 Test tools 8](#_Toc146700424)

[1.7 Test Result 8](#_Toc146700425)

[2 Vendor, system under test (SUT) and test result information 10](#_Toc146700426)

[2.1 Information about the vendor 10](#_Toc146700427)

[2.2 Information about system under test (SUT) 10](#_Toc146700428)

[2.3 Information about the test result 10](#_Toc146700429)

[3 The test 11](#_Toc146700430)

[3.1 Documentation of the test 11](#_Toc146700431)

[3.2 Test of TouchStone testscripts 12](#_Toc146700432)

[3.3 Test of requirements for content and flow/workflows 13](#_Toc146700433)

[3.4 Test of general technical requirements 29](#_Toc146700434)

# Introduction

This is a test protocol for receiving a HospitalNotification.

All documentation concerning HospitalNotification and Governance (se [Background](#_Baggrundsmaterialer) material) will be the subject 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-version of the standard but may have a patch version that is different from the standard’s patch-version.

**Regarding sending an Acknowledgement:** Approval requires that the system under test (SUT) is approved for sending FHIR Acknowledgements (DK: kvittering). This test is handled in a separate test protocol. However, when testing the receipt of HospitalNotification, SUT will have to demonstrate that SUT generates and sends a FHIR receipt in response to a received HospitalNotification (see more below).

**Regarding receipt of HospitalNotification and sending an Acknowledgement though the municipal message distributor (DK: Fælleskommunal Beskedfordeler):** HospitalNotification is received through the municipal message distributor and will be wrapped in a message distributor envelope (DK: beskedfordelerkuvert) for receipt (DK: Kvittering). The receiver system must acknowledge with a FHIR receipt (DK: Kvittering) which is also sent through the municipal message distributor (DK: Fælleskommunal Beskedfordeler) and must be packaged in a message distributor envelope. In connection with testing the reception of HospitalNotification, SUT will have to demonstrate that SUT generates and sends a FHIR receipt in response to a received HospitalNotification, but the actual test of reception and sending HospitalNotification in a message distribution envelope is done within the framework of KOMBIT.

## 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 vendor carries 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:
   * [Clinical guidelines for application](bookmark://_Baggrundsmaterialer)
   * [Use cases](bookmark://_Baggrundsmaterialer)
   * [Implementation Guide](bookmark://_Baggrundsmaterialer)
   * Governance
   * And other relevant materials, cf. the [Background material](#_Background_materials).
2. The vendor has performed [self-test](bookmark://Egentest), approved by MedCom.
3. The vendor has created the [relevant test persons](#_Testeksempler_og_testpersoner) in system under test (SUT). Please note that one of these test persons must be a citizen without relevant cases in the SUT’s case complex, i.e. a citizen who does not have an active case within the areas that give legal authority to receive a HospitalNotification (see [clinical guidelines for application](#_Baggrundsmaterialer) ). This could be a citizen who only receives assistive technology.
4. The vendor uses the same version of SUT during self-test and live test.
5. Approval requires that the SUT is approved for sending FHIR-Acknowledgement (DK: kvittering).

## Documentation of self-test

**Self-test**

**Prior to the test, the vendor must have performed self-test, including successfully completed TouchStone self-tests, which are 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:

* [Current 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 receiver (R) of the standard
* File type

Example: HospitalNotification\_3.4\_A\_R.xml or HospitalNotification\_2.2\_B\_R.png

## Background materials

| **Name** | **Version[[1]](#footnote-2)** | **Link/reference** | **Description** |
| --- | --- | --- | --- |
| HospitalNotification – documentation site | 3.0.X | [https://medcomdk.github.io/dk-medcom-HospitalNotification/](https://medcomdk.github.io/dk-medcom-hospitalnotification/) | Documentation site with references to all relevant documentation, including:   * Clinical guidelines for application * Use cases * Technical specifications |
| Implementation Guide | 3.0.X | [https://medcomfhir.dk/ig/HospitalNotification/](https://medcomfhir.dk/ig/hospitalnotification/) |  |
| Governance for MedCom FHIR | 1.0.0 | <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 the message |
| SOP for MedCom’s test and certification | 2.7 | <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. |

## Test examples and test persons

|  |  |  |
| --- | --- | --- |
| **Name** | **Link/reference** | **Description** |
| Test examples /FHIR example files | <http://medcomfhir.dk/ig/hospitalnotificationtestscripts>/ testexamples.html | Includes test examples and an overview of all these. |
| Overview of test persons | <https://www.medcom.dk/opslag/koder-tabeller-ydere/tabeller/nationale-test-cpr-numre> | Overview of national test personal identification number (DK:CPR-nummer), that can be used during test.  **Note:** During test and certification, 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/> | Public server that validates against MedCom's FHIR profiles. It is permitted to use the server for testing the upload/download of FHIR resources. |
| TouchStone | <https://touchstone.aegis.net/touchstone/> | Test tool for testing the FHIR standard.  The vendor can get access to TouchStone as an organisation - either through a license that MedCom supplies (inquiry at [fhir@medcom.dk](mailto:fhir@medcom.dk)), or a license that the supplier has acquired itself.  Find [instructions for TouchStone](https://medcomdk.github.io/MedComLandingPage/assets/documents/TouchStoneGettingStarted.html) here |
| TouchStone test scripts | Link to IG including test scripts and instructions for usage hereof:  <http://medcomfhir.dk/ig/hospitalnotificationtestscripts>/ testexamples.html  Direct link to the test scripts in TouchStone: [https://touchstone.aegis.net/touchstone/testdefinitions? selectedTestGrp=%2FFHIRSandbox%2FMedCom%2FHospitalNotification& activeOnly=false&includeInactive=false&ps=50&sb=qualifiedName&sd= DESC&allSelected=false&contentEntry=ALL](https://touchstone.aegis.net/touchstone/testdefinitions?%20selectedTestGrp=%2FFHIRSandbox%2FMedCom%2FHospitalNotification&%20activeOnly=false&includeInactive=false&ps=50&sb=qualifiedName&sd=%20DESC&allSelected=false&contentEntry=ALL) | Test scripts relevant for the standard.  Find [instructions for TouchStone](https://medcomdk.github.io/MedComLandingPage/assets/documents/TouchStoneGettingStarted.html) here |

## Test Result

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

| **Marking** | **F1** | **F2** | **F3** | **F4** | **Ok** |
| --- | --- | --- | --- | --- | --- |
| **Evaluation** | **Critical** | **Serious** | **Significant** | **Less significant** | **Approved** |

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

Approval requires that SUT is approved for receiving FHIR Acknowledgement (DK: Kvittering)

For further information, please read: [MedCom’s test and](#TestCertificering) certification.

# 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 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 the test result

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

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

# 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 testscripts
2. Test of requirements for content and flow/workflows, including received receipts
3. Test of technical requirements

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

## Documentation of the test

**Documentation of the test**

As valid documentation, the test participant or test manager must document completion by continuous screen dumps (.png/.jpeg) and/or files/log files (.xml/.json). **Before the test, it is agreed who is responsible for this.**

The following applies:

* The files must be viewable in a standard tool and must not require further processing by MedCom
* 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 receiver (R) of the standard
  + File type

Example: HospitalNotification\_3.4\_A\_R.xml or HospitalNotification\_2.2\_B\_R.png

If the vendor has documented the test themselves, the files must be sent in a ZIP file to [fhir@medcom.dk](mailto:fhir@medcom.dk).

## Test of TouchStone testscripts

The purpose of this test step is to ensure that SUT can read HospitalNotification correct and complies with the rules in the [Implementation Guide](#_Baggrundsmaterialer). This test step is optional to execute.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data/test person** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | **Optional test step:**  Run all test scripts for use cases and user flows in TouchStone. |  | All test scripts completed without errors. |  | Choose |

## Test of requirements for content and flow/workflows

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

It is up to the receiver system (and customers) to establish rules for any automatic pause of municipal services upon receipt of HospitalNotification.

It is up to the receiver system (and customer), how the end user is notified/made aware of receipt of HospitalNotification in the user interface.

The raw data file (with its entire content, including an indication of whether an admission report must be sent) must always be accessible by a technical user. Selected data must (as a minimum) be visible to SUT users in the user interface. If the sections appear with a heading in the user interface, it is recommended to use headings as defined in Table 1. The message must always - on the user interface - be named as "HospitalNotification".

|  |  |  |  |
| --- | --- | --- | --- |
| **The section headline** | **Content** | **Example** | |
| Borger | Citizens personal identification number | Borger | XX-XX-XXX |
| Status for sygehusophold | Status of the citizen’s hospital stay (type of HospitalNotification). See [Background materials](#_Baggrundsmaterialer) . | Status for sygehusophold | Start sygehusophold – indlagt |
| Tidspunkt for sygehusophold | The date and time, when the citizen was admitted/went on leave, ended his leave, was discharged, died, etc. | Tidspunkt | 22-08-18 kl. 9:30 |
| Hospital | The Name of the Hospital where the citizen is admitted. | Hospital | Odense Universitetshospital |
| Afdeling | The name of the department where the citizen is admitted | Afdeling | Infektionsmedicinsk afdeling |
| Afsnit | The name of the unit where the citizen is admitted | Afsnit | Q |

Table 1 Minimum requirements for sections to be displayed to the end user, as a recommendation for section names.

The table below reflects the use cases that are tested in relation to content and flow/workflows. The table also shows which example files must be loaded in connection with each test step. An overview of the [example fil](#_Testeksempler_og_testpersoner)es can be found in [Appendix I](#BilagI).

In the individual test steps, a reference is made to of the following HospitalNotifications types: [STIN], [STAA], [STOR], [SLOR], [SLHJ], [MORS], [AN\_XX], [RE\_XX]. The description of the HospitalNotifications types is available on GitHub (HospitalNotification Codes), see [Background materials.](#_Baggrundsmaterialer)

| [**Use case**](#_Baggrundsmaterialer) | **Description** | **Section** | **Example file** |
| --- | --- | --- | --- |
| R1 | The citizen is **admitted** without prior admission | 3.3.1 | [STIN\_A] |
| R1 | The citizen is **admitted** after prior admission to another hospital in another region (transfer between regions) | 3.3.1 | [STIN\_B1]  [STIN\_B2] |
| R1.A1 | The citizen is **admitted** after a prior admission to another hospital in the same region (transfer between hospitals in the same region) | 3.3.2 | [STIN\_C1]  [STIN\_C2] |
| R2 | The citizen is referred to an acute ambulant hospital stay | 3.3.3 | [STAA\_D] |
| R1.A1 | The citizen is **admitted** after a previous stay in the outpatient department at the same hospital | 3.3.4 | [STAA\_E]  [STIN\_E] |
| R3 | The citizen goes on **leave** from his/her hospital stay | 3.3.5 | [STIN\_F]  [STOR\_F] |
| R4 | The citizen returns at the hospital after **leave** | 3.3.6 | [STIN\_G]  [STOR\_G]  [SLOR\_G] |
| R6 | The citizen is **discharged** | 3.3.7 | [STIN\_H]  [SLHJ\_H] |
| R6 | The citizen is **discharged** because the citizen has not returned to the hospital after the end of leave | 3.3.7 | [STIN\_I]  [STOR\_I]  [SLHJ\_I] |
| R7 | The citizen **dies** upon arrival | 3.3.8 | [MORS\_J] |
| R7 | The citizen **dies** during the hospital stay | 3.3.8 | [STIN\_K]  [MORS\_K] |
| R7 | The citizen **dies** during leave | 3.3.8 | [STIN\_L]  [STOR\_L]  [MORS\_L] |
| R8 | **Manual end** of hospital stay in the receiver system | 3.3.9 | [STIN\_M] |
| R.CANC | Receipt of **cancellation** (wrong citizen/ Person identification number or wrong type of hospital stay) | 3.3.11 | [STIN\_N]  [AN\_STIN\_N] |
| R.CORR | Receipt of **correction** (wrong time) | 3.3.10 | [STIN\_O]  [RE\_STIN\_O] |
| R.CORR | Receipt of **correction** (wrong hospital department) | 3.3.10 | [STIN\_P]  [RE\_STIN\_P] |

### Use case R1: Admission/receipt of [STIN] with request for an admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **Admission of a citizen**  Load test data and show that the SUT user can see that HospitalNotification of the type “Start sygehusophold - indlagt” has been received. | [STIN\_A] | The SUT user can see that a HospitalNotification of the type ”Start sygehusophold - indlagt”  SUT-user can (as minimum) see the following:   * The personal identification number (CPR-nr.) * Status for hospital stay: ”Start sygehusophold - indlagt” * Date and time of hospital stay (Encounter.period.start) * Name of the hospital department /section and where the citizen is admitted   The SUT user can see that the citizen appears admitted/absent due to hospital stay. |  | Choose |
|  | Demonstrate that SUT has sent an automatic admission report (XDIS16) on the citizen and that this is visible to the SUT- user |  | An automatic admission report (XDIS16) has been sent in response to [STIN\_A]. This is visible to the SUT user. |  | Choose |
|  | **The citizen is admitted to a new hospital in another region**  Load the test data files.  Demonstrate that the SUT user can see, that a HospitalNotification of type: ”Start sygehusophold - indlagt” has been received and a HospitalNotification of type: ”Start sygehusophold - indlagt” form another hospital has been received | [STIN\_B1]  [STIN\_B2] | The SUT user can see that a HospitalNotification of type ”Start sygehusophold - indlagt” has been received, then a HospitalNotification of type: ”Start sygehusophold - indlagt” form another hospital form another region has been received  The SUT user can see that the citizen appears admitted/absent due to hospital stay. |  | Choose |
|  | Demonstrate, that SUT, has automatically sent an admission note (XDIS16), on citizen, and that this is visible to SUT-users. |  | Two automatic admission notes (XDIS16) have been sent in response to both HospitalNotifications. This is visible to the SUT user. |  | Choose |

### Use case R1.A1: Citizen admitted/receipt of [STIN] without request for admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **The citizen is admitted to a new hospital in the same region.**  Load the test data files  Demonstrate that the SUT user can see that there is first received a HospitalNotification of type ”Start sygehusophold - indlagt”, then a HospitalNotification of type “Start sygehusophold - indlagt” from another hospital. | [STIN\_C1]  [STIN\_C2] | The SUT user can see that first a HospitalNotification of type: ”Start sygehusophold - indlagt” has been received, then a HospitalNotification and ”Start sygehusophold - indlagt” form another hospital in the same region has been received.  The SUT user can see that the citizen appears admitted/absent due to hospital stay. |  | Choose |
|  | Demonstrate that SUT, has not sent an automatic admission note in response to [STIN\_C2] |  | An automatic admission note (XDIS16) has not been sent in response to [STIN\_C2]. This is visible to the SUT user. |  | Choose |

### 

### Use case R2: Citizen is an acute ambulant patient at the hospital/receipt of [STAA] with request for admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load the test data and demonstrate, that the SUT user can see, that a HospitalNotification of the type: ” Start sygehusophold - akut ambulant ” has been received. | [STAA\_D] | The SUT user can see that a HospitalNotification of type ” Start sygehusophold - akut ambulant” has been received.  Date and time for hospital stay = Encounter.period.start  The SUT user can see that the citizen appears absent due to hospital stay. |  | Choose |
|  | Demonstrate that the SUT has sent an automatic admission note (XDIS16) to the citizen and that this is visible to SUT-user |  | An automatic admission note (XDIS16) has been sent in response to [STAA\_D]. This is visible to the SUT user. |  | Choose |

### Use case R1.A1: A citizen is admitted after a prior acute ambulant hospital stay/receipt of [STIN] without request for admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data files.  Demonstrate that the SUT user can see that there is first received a HospitalNotification of type “Start sygehusophold - akut ambulant”, then a Hospital Notification of type “Start sygehusophold - indlagt”. | [STAA\_E]  [STIN\_E] | The SUT user can see that first a HospitalNotification of type “Start sygehusophold - akut ambulant ” has been received, then a HospitalNotification of type “Start sygehusophold - indlagt”.  The SUT user can see that the citizen appears admitted/absent due to hospital stay. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to [STINE\_E]. This is visible to the SUT user. |  | Choose |

### 

### Use case R3: Citizen goes on leave from his hospital stay/receipt of [STOR] without requesting admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom admission** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data files  Demonstrate that the SUT user can see that there is first received a HospitalNotification of type : “Start sygehusophold - indlagt” , then a HospitalNotification of the type “Start orlov” | [STIN\_F]  [STOR\_F] | The SUT user can see that first HospitalNotification of type: “Start sygehusophold - indlagt” has been received, then a HospitalNotification of the type “Start orlov”.  Date and time for hospital stay/ incident = Encounter.extension:leavePeriod.start.  The SUT user can see that the citizen appears admitted/absent due to hospital stay. It may appear that the citizen is on leave. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to [STOR\_F]. This is visible to the SUT user. |  | Choose |

### Use case R4: Citizen returns to hospital form leave/receipt of [SLOR] without requesting admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data files  Demonstrate, that SUT user can see that there is first received a HospitalNotification of type: “Start sygehusophold - indlagt”, then a HospitalNotification of the type “Start orlov” and lastly a HospitalNotification of type “ Slut orlov” | [STIN\_G]  [STOR\_G]  [SLOR\_G] | The SUT user can see, that first HospitalNotification of type: “Start sygehusophold - indlagt” has been received, then a HospitalNotification of the type “Start orlov” and lastly a HospitalNotification of type “Slut orlov”.  Date and time for hospital stay/ incident = Encounter.extension:leavePeriod.end  The SUT user can see that the citizen appears admitted/absent due to hospital stay. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to either [STOR\_G] or [SLOR\_G]. This is visible to the SUT-user. |  | Choose |

### Use case R6: Citizen discharged / receipt of [SLHJ] without requesting an admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data files  Demonstrate that the SUT user can see that there is first received a HospitalNotification of type “Start sygehusophold - indlagt”, then a HospitalNotification of type “End hospital stay – patient discharged and sent home/primary sector”. | [STIN\_H]  [SLHJ\_H] | The SUT user can see, that first HospitalNotification of type: “Start sygehusophold - indlagt” has been received, then a HospitalNotification of the type “Slut sygehusophold - afsluttet til hjemmet/primær sektor”.  Date and time for hospital stay/ incident = Encounter.period.end  The SUT user can see that the citizen appears as ‘active’ again, as the citizen has been discharged. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to [SLHJ\_H]. This is visible to the SUT user. |  | Choose |
|  | **Discharge of patient, who is absent after leave**  Load test data-files.  Demonstrate that the SUT-user can see that there is first received a HospitalNotification of type “Start sygehusophold - indlagt”, then a HospitalNotification of type “Start orlov” and lastly a HospitalNotification of type “End hospital stay – patient discharged and sent home/primary sector”. | [STIN\_I]  [STOR\_I]  [SLHJ\_I] | The SUT user can see, that first HospitalNotification of type: “Start sygehusophold - indlagt” has been received, then a HospitalNotification of the type “Start orlov“ and lastly a HospitalNotification of type “Slut sygehusophold - afsluttet til hjemmet/primær sektor”.  The SUT user can see that the citizen appears as ‘active’ again, as the citizen has been discharged. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to either [STOR\_I] or [SLHJ\_H]. This is visible to the SUT user. |  | Choose |

### 

### Use case R6: Citizen dies/ receipt of [MORS] without requesting an admission note (XDIS16)

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data and show that the SUT user can see that a Hospital notification of type „Død“ has been received | [MORS\_J] | The SUT user can see that a HospitalNotification of type ”Død” has been received.  Date and time for hospital stay/ incident = Encounter.period.end  The SUT user can see that the citizen appears dead. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to [MORS\_J]. This is visible to the SUT user. |  | Choose |
|  | **Citizen dies during hospital stay**  Load test data files.  Demonstrate that the SUT user can see that a Hospital notification of type “Start sygehusophold - indlagt”, has been received and then a HospitalNotification of type“ Død“ | [STIN\_K]  [MORS\_K] | The SUT user can see that a HospitalNotification of type ” Start sygehusophold - indlagt ” has been received, then a HospitalNotification of type „Død“  The SUT user can see that the citizen appears dead. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to [MORS\_K]. This is visible to the SUT user. |  | Choose |
|  | **Citizen dies during on leave**  Load test data files.    Demonstrate that the SUT user can see that a Hospital notification of type “Start sygehusophold - indlagt”, has been received, then a HospitalNotification of type “Start orlov “ and lastly a HospitalNotification of type“ Død“ | [STIN\_L]  [STOR\_L]  [MORS\_L] | The SUT user can see that a HospitalNotification of type ” Start sygehusophold - indlagt ” then a HospitalNotification of type“ Start orlov“ and lastly a HospitalNotification of type „Død“ has been received.  The SUT user can see that the citizen appears dead. |  | Choose |
|  | Demonstrate that the SUT has not sent an automatic admission note (XDIS16) on the citizen. |  | An automatic admission note (XDIS16) has not been sent in response to either [STOR\_L] or [MORS\_J]. This is visible to the SUT user. |  | Choose |

### Manual end of hospital stay in receiver system

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data file  Demonstrate that the SUT user can manually end a citizen’s stay at the hospital because of absence of HospitalNotification type [SLHJ] | [STIN\_M] | The SUT user has manually ended the citizen’s stay at the hospital.  The SUT user can see that the citizen appears as ‘active’ again. |  | Vælg |

### Use case R.CORR: Correction/receipt of [RE\_XX]

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **Correction to time**  Load test data files.  If implemented, demonstrate how user, is notified that a HospitalNotification of type [RE\_STIN] has been received as a correction to an already received HospitalNotification .  Show how the SUT user can subsequently see what has been corrected. | [STIN\_O]  [RE\_STIN\_O] | If relevant, the SUT user is notified that a correction has been received for [STIN\_O].  The SUT user can see the correction (time) that has been made (track changes) and can access and see the corrected time and hour for the start of the hospital stay |  | Choose |
|  | Show that SUT has not sent an automatic admission report (XDIS16) in response to [RE\_STIN\_O] |  | An automatic admission report (XDIS16) has not been sent in response to [RE\_STIN\_O]. This is visible to the SUT user. |  | Choose |
|  | **Correction to hospital department**  Load the test data files.  If implemented, demonstrate how the SUT user, is notified that a HospitalNotification of type [RE\_STIN] has been received as a correction to an already received HospitalNotification, and show how SUT user can subsequently see what has been corrected. | [STIN\_P]  [RE\_STIN\_P] | If relevant, the SUT user is notified that a correction has been received for [STIN\_P].  The SUT user can see the change (hospital department) that has been made (change flag) and can access the corrected information. |  | Choose |
|  | Show that SUT has not sent an automatic admission report (XDIS16) in response to [RE\_STIN\_P] |  | An automatic admission report (XDIS16) has not been sent in response to [RE\_STIN\_P]. This is visible to the SUT user. |  | Choose |

### Use case R.CANC: Cancellation /receipt of [AN\_XX]

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data files.  Demonstrate how the SUT user is notified that a HospitalNotification of type [AN\_STIN] has been received as a cancellation to an already received HospitalNotification.  Show how the SUT user can subsequently see that the original HospitalNotification has been cancelled. | [STIN\_N]  [AN\_STIN\_N] | The SUT user is notified that a cancellation has been received for HospitalNotification of Type [STIN\_N] .  The SUT user can see that the previously received HospitalNotification has been cancelled (cancellation view)  The SUT user can see that the citizen appears as ‘active’ again. |  | Choose |
|  | Show that the SUT has not sent an automatic admission report (XDIS16) in response to [AN\_STIN\_N] |  | An automatic admission report (XDIS16) has not been sent in response to [AN\_STIN\_N]. This is visible to the SUT-user. |  | Choose |
|  | **Cancellation after correction**  Load test data which is a cancellation to a HospitalNotification [STIN\_P] that has been corrected [RE\_STIN\_P] (from test step 3.3.10.3)  Demonstrate that the SUT user can clearly see that the original HospitalNotification has been cancelled, and that the citizen is not staying at the hospital | [AN\_STIN\_P] | The SUT user can see that the previously received HospitalNotification has been cancelled (cancellation view).  The SUT user can see that the citizen appears as ‘active’ again (not admitted). |  |  |

### Naming and presentation of sections in the user interface

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load a random test data file and show how this is presented to the SUT user. |  | The file is loaded and available to the SUT user. |  | Choose |
|  | Demonstrate that the SUT user can see minimum information cf. Table 1.  Note that SUT – if the Organization.name element is not sent along – must make an entry in the SOR register to ensure that information about where citizen is hospitalized can be displayed to SUT user. |  | As a minimum, SUT users can see:   * (Citizen’s) personal identification number (CPR no.) * Status of hospital stay * Time of hospital stay * Hospital * Department * Unit |  | Choose |
|  | Demonstrate that the message appears with the correct name=” Advis om sygehusophold”. |  | The message appears with the name ”Advis om sygehusophold” |  | Choose |
|  | If the message appears with section headings, show that headings are used as defined in Table 1 (recommendation) or equivalent headings are used that are indicative of the content and without risk of misunderstanding. |  | The sections appear with headings as defined in Table 1 (recommended) or with headings that are indicative of the content and with no risk of being misunderstood. |  | Choose |

## Test of general technical requirements

The purpose of these test steps is to ensure that the technical receipt of HospitalNotification is implemented with satisfactory quality, i.e. supports governance for message communication at a general level, as well as governance for HospitalNotification as described in 1.4.

### Receipt of HospitalNotification and sending of Acknowledgement

| **Test step #** | **Action** | **Test data** | **Expected results** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Describe how FHIR-messages are loaded into the SUT.  For example: How is data loaded into SUT – via. Mapping to internal format or to local FHIR infrastructure? |  | For example: FHIR messages are used directly or processed. |  | Choose |
|  | Run all TouchStone test scripts that are relevant for sending an Acknowledgement, based on a received HospitalNotification:  **Note:** See information here [IG including a description of relevant test scripts](https://medcomfhir.dk/ig/acknowledgementtestscripts/testscripts.html#send-acknowledgement-test-scripts), and the test suite here: [TouchStone test suite for sending Acknowledgements](https://touchstone.aegis.net/touchstone/conformance/current?suite=FHIR4-0-1-Acknowledgement-v200-Send-Client) |  | All relevant test scripts are completed without errors. |  | Choose |
|  | Load test data and demonstrate that SUT returns a FHIR Acknowledgement. | FHIR example file [TEK\_STIN\_A] | SUT returns a FHIR Acknowledgement |  | Choose |
|  | Find and open any sent XDIS16 message sent in response to a received HospitalNotification.  Verify that the XDIS16 sent is correctly formatted and that data from the received HospitalNotification is correctly transferred to the XDIS16 message. |  | XDIS16 is correctly parsed in the validation tool “<https://xml.medcom.dk>”  XDIS16 has been sent to the correct recipient as indicated in the HospitalNotification message.  The time stamps on XDIS16 are later than the HospitalNotification that triggered the action. |  | Choose |
|  | Describe or demonstrate how receiving an [OIOXML Acknowledgement](https://svn.medcom.dk/svn/releases/Standarder/Den%20gode%20CONTRL/XML/Dokumentation/)  of type ’XCTL01’, ’XCTL02’ and ’XCTL03’ is handled in the SUT., i.e. both positive and negative receipts.  Note: This is the Acknowledgement to a sent admission note. |  | Description or screenshot of how OIOXML Acknowledgement of type ’XCTL01’, ’XCTL02’and ’XCTL03’ is received in SUT. |  | Choose |

### Use of EpisodeOfCareIdentifier

| **Test step #** | **Action** | **Test data** | **Expected results** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | **Use of EpisodeOfCareIdentifier (locally defined UUID)**  Select any loaded test data file as well as the following sent XDIS16 message.  Demonstrate that the EpisodeOfCareIdentifier in the two messages is identical\* and that it is clear to the SUT user that the two messages are related.  \*Note that the EpisodeOfCareIdentifier in the XDIS16 will be without hyphens, while the EpisodeOfCareIdentifier in the HospitalNotification is hyphenated. | FHIR example file [TEK\_ID\_LOCAL] | SUT is able to load a HospitalNotification message where EpisodeOfCareIdentifier is a locally defined UUID. SUT returns the locally defined UUID in the XDIS16 message (EpisodeofCareIdentifier is identical in the two messages) It is clear to the SUT user that the two messages are related. |  | Choose |
|  | **Use of EpisodeOfCareIdentifier (LPR3 identifier**)  Load test data and send XDIS16. The test data file contains an LPR3 identifier. | FHIR example file [TEK\_ID\_LPR3] | SUT is able to load a HospitalNotification where the EpisodeOfCareIdentifier is an LPR3 identifier. |  | Choose |
|  | Show that SUT returns the LPR3 identifier in the generated XDIS16 message and that it is clear to the SUT user that the two messages are related. |  | SUT returns the LPR3 identifier in the XDIS16 message.  It is clear to the SUT user that the two messages are related. |  | Choose |
|  | **Use of EpisodeOfCareIdentifier (N > 1)**  Load test data and send XDIS16. The test data file contains two EpisodeOfCareIdentifiers (a locally defined UUID and an LPR3 identifier). | FHIR example file [TEK\_ID\_2] | SUT is able to load a HospitalNotification message with more than one EpisodeOfCareIdentifer. |  | Choose |
|  | Demonstrate that SUT returns the locally defined UUID as EpisodeOfCareidentifer in the generated XDIS16 message, and that it is clear to the SUT user that the two messages are related |  | SUT returns the locally defined UUID.  It is clear to the SUT user that the two messages are related. |  | Choose |

### Loading series of FHIR messages in an admission flow, where receipt of messages does not follow the order in which the messages have been generated (sent/received).

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load the test data and demonstrate that the citizen appears as absent in the receiving system due to hospitalization. | FHIR example file [TEK\_STAA\_A] | The SUT user sees that the citizen appears as absent due to hospitalization. SUT sends a FHIR Acknowledgement back to the correct recipient. |  | Choose |
|  | Load test data and demonstrate that the citizen appears as ‘active’/’completed’/’discharged’ from the hospital stay.  Save both the FHIR message received and the acknowledgement sent. | FHIR example file [TEK\_SLHJ\_A] | The SUT user sees that the citizen appears as ‘active’ from the hospital stay  SUT sends a FHIR Acknowledgement back to the correct recipient. |  | Choose |
|  | Load test data and demonstrate that the citizen appears as ‘active’/’completed’/’discharged’ from the hospital stay, but that the message history shows that a HospitalNotification [STIN] was received.  Test step tests, that the receiving system shows correct status for the citizen, even if messages are not sent or received in the order in which the events occurred.  Save both the FHIR message received, and the Acknowledgement sent. | FHIR example file [TEK\_STIN\_B] | The SUT user sees that the citizen appears as ‘active’ from the hospital stay even though a notification of admission has subsequently been received. The SUT sends a FHIR receipt (Acknowledgement) back to the correct recipient, even if the message is only loaded as history. |  | Choose |

### Handling duplicates

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data and demonstrate that the HospitalNotification is loaded and available to the SUT user | FHIR example file [TEK\_DUB] | The SUT user can see that a HospitalNotification of type ” Start sygehusophold - indlagt” has been received. |  | Choose |
|  | Demonstrate that SUT has sent a positive FHIR Acknowledgement. |  | The SUT has positively acknowledged the HospitalNotification and sent a FHIR Acknowledgement to the correct recipient. |  | Choose |
|  | Load test data again, and demonstrate that the HospitalNotification, which is a duplicate, is ignored, that the SUT user can still only see one received HospitalNotification of type [STIN], and that no XDIS16 message has been sent again. | FHIR example file [TEK\_DUB] | The SUT user can still only see that one HospitalNotification has been received of the type: “Start sygehusophold - indlagt”  No XDIS16 message has been sent again. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR acknowledgement for the duplicate. |  | SUT has positively acknowledged the HospitalNotification and sent a FHIR Acknowledgement to the correct recipient. |  | Choose |

### Handling of HospitalNotification for a citizen who does not have a relevant case in the SUT

| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
| --- | --- | --- | --- | --- | --- |
|  | Load test data and demonstrate how the SUT handles receiving a HospitalNotification for a citizen who does not have a relevant case in the SUT. | FHIR example file  [TEK\_LOV] | HospitalNotification is not loaded and is not presented to the SUT user. |  | Choose |
|  | Demonstrate that the SUT has sent a positive FHIR Acknowledgement. |  | SUT has positively acknowledged the HospitalNotification and sent a FHIR Acknowledgement to the correct recipient. |  | Choose |
|  | Demonstrate that the SUT has logged receipt of the HospitalNotification on a citizen who does not have a case in the SUT. |  | The SUT demonstrates - via a log of received HospitalNotification messages for citizens without relevant cases in the SUT – that receipt of the message has been logged. |  | Choose |

### Handling of messages containing errors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | Load test data and demonstrate how SUT handles missing valid ”Bundle.entry.resource.ofType(Encounter).class.code” | FHIR example file  [TEK\_FCC] | The SUT returns a MedCom Acknowledgement describing the error. |  | Choose |
|  | The SUT must not display the message containing errors. |  | SUT does not display the message containing errors. |  | Choose |

## 

### Handling of HospitalNotifcation in XML and JSON formats

### 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test step #** | **Action** | **Test data** | **Expected result** | **Actual result** | **MedCom assessment** |
|  | **Format of the HospitalNotification Message**  Demonstrate that SUT can handle a HospitalNotification message as an XML and JSON file.  Load the test file with HospitalNotifcation message in XML format.  Load the test file with HospitalNotification message in JSON format. |  | The SUT loads correctly test files with an HospitalNotification in XML and JSON format. |  | Choose |

1. X expresses patch-level versioning, which includes minor fixes that are backward compatible. [↑](#footnote-ref-2)