European Security in Health Data Exchange

Deliverable D5.3

Consent Management Tool

<table>
<thead>
<tr>
<th>Editor(s):</th>
<th>Niall Burns, Ewan Dawson, Graham Davidson</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsible Partner:</td>
<td>Symphonic</td>
</tr>
<tr>
<td>Status-Version:</td>
<td>Final</td>
</tr>
<tr>
<td>Date:</td>
<td>28/12/2017</td>
</tr>
<tr>
<td>Distribution level (CO, PU):</td>
<td>CO</td>
</tr>
</tbody>
</table>

Project Title: SHIELD

Contract No. GA 727301

http://project-shield.eu/

Page 1 of 36
<table>
<thead>
<tr>
<th><strong>Project Number:</strong></th>
<th>GA 727301</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Title:</strong></td>
<td>SHIELD</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Title of Deliverable:</strong></th>
<th>Consent management (M7-M30, Leader: Symphonic, Participants: IBM)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Due Date of Delivery to the EC:</strong></td>
<td>31/12/2017</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Workpackage responsible for the Deliverable:</strong></th>
<th>5.3</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Editor(s):</strong></th>
<th>Ewan Dawson, Niall Burns, Graham Davidson</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Contributor(s):</strong></th>
<th>Ewan Dawson, Niall Burns, Graham Davidson</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Reviewer(s):</strong></th>
<th>Micha Moffie, Xabier Larrucea</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Approved by:</strong></th>
<th>All WP’s</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Recommended/mandatory readers:</strong></th>
<th>All WP’s</th>
</tr>
</thead>
</table>

**Abstract:**
This task will identify the mechanisms for capturing patient consent and access preferences and integrate these into the access model to ensure that these reflect the balanced concerns of patient privacy and treatment need. It will consider handling of “steady state” consent models with the requirements and treatment of override circumstances, such as “break-glass” emergency or out-of-bounds scenarios. Consideration will be given to the integration of notification to patients and relevant clinicians when out-of-bounds events require exceptions to steady state access rules. This will align consent with the data hiding and anonymization techniques and tools in Task 5.4.

<table>
<thead>
<tr>
<th><strong>Keyword List:</strong></th>
<th>Consent management, decision engine, enforcement</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Disclaimer</strong></th>
<th>This document reflects only the author’s views and neither Agency nor the Commission are</th>
</tr>
</thead>
</table>
responsible for any use that may be made of the information contained therein
# Document Description

## Document Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modifications Introduced</th>
<th>Modification Reason</th>
<th>Modified by</th>
</tr>
</thead>
<tbody>
<tr>
<td>v0.1</td>
<td>26/09/2017</td>
<td>First draft ToC version</td>
<td>Niall Burns, Ewan Dawson (Symphonic)</td>
<td></td>
</tr>
<tr>
<td>V0.2</td>
<td>29/11/2017</td>
<td>Second Draft</td>
<td>Graham Davidson</td>
<td></td>
</tr>
<tr>
<td>V0.3</td>
<td>01/12/2017</td>
<td>Third Draft</td>
<td>Graham Davidson</td>
<td></td>
</tr>
<tr>
<td>V0.4</td>
<td>05/12/2017</td>
<td>Review draft</td>
<td>Graham Davidson, Niall Burns, Ewan Dawson</td>
<td></td>
</tr>
<tr>
<td>V0.5</td>
<td>20/12/2017</td>
<td>Post review modifications</td>
<td>Niall Burns</td>
<td></td>
</tr>
<tr>
<td>V1.0</td>
<td>28/12/2017</td>
<td>Final</td>
<td>Xabier Larrucea</td>
<td></td>
</tr>
</tbody>
</table>
Table of Contents

Table of Contents ........................................................................................................................................ 5
Terms and abbreviations............................................................................................................................. 7
Executive Summary ....................................................................................................................................... 8
1 Introduction .................................................................................................................................................. 9
  1.1 Objective and Scope.................................................................................................................................. 9
  1.2 Document Structure ............................................................................................................................... 9
2 Consent Management .................................................................................................................................. 10
  2.1 Overview .................................................................................................................................................. 10
  2.2 Data Model ............................................................................................................................................. 10
  2.3 API .......................................................................................................................................................... 10
    2.3.1 GET /api/consent (List all consents for a specific patient) ............................................................. 11
    2.3.2 GET /api/consent/{documentId} (Show consents for a specific document) ....................... 11
    2.3.3 POST /api/consent (Create a consent record) ........................................................................... 12
    2.3.4 PUT /api/consent/{id} (Update a consent record) ................................................................. 12
    2.3.5 DELETE /api/consent/{id} (Delete a consent record) ....................................................... 13
3 Use Case Requirement Analysis .............................................................................................................. 14
  3.1 Use Case 1 – ‘Break Glass’ Scenario .................................................................................................. 14
    3.1.1 Key components Diagram ............................................................................................................. 14
    3.1.2 Sequence Diagram - Document Listing ..................................................................................... 15
    3.1.3 Sequence Diagram - Document Viewing .................................................................................... 16
    3.1.4 Enforcement Point ......................................................................................................................... 17
    3.1.5 Decision Engine Policies ............................................................................................................. 18
  3.2 Use Case 2: Surgical Intervention ....................................................................................................... 19
    3.2.1 Key components Diagram ............................................................................................................. 19
    3.2.2 Sequence Diagram - Consent Update via Mobile Device ....................................................... 20
    3.2.3 Sequence Diagram - Document Listing ..................................................................................... 21
    3.2.4 Sequence Diagram - Document Viewing .................................................................................... 22
    3.2.5 Enforcement Point ......................................................................................................................... 23
    3.2.6 Decision Engine Policies ............................................................................................................. 24
  3.3 Use Case 3: Chronic Conditions + Remote Monitoring ........................................................................ 26
    3.3.1 Data Flow Diagram ....................................................................................................................... 26
    3.3.2 Sequence Diagram – Document Listing Multiple Jurisdictions ........................................... 27
    3.3.3 Sequence Diagram - Document Viewing .................................................................................... 28
    3.3.4 Enforcement Point ......................................................................................................................... 29
    3.3.5 Decision Engine Policies ............................................................................................................. 31
4 Consent Management Tools requirements .............................................................. 32
   TRUST ................................................................................................................... 32
   GOVERNANCE ...................................................................................................... 32
   GATEWAY ................................................................................................................ 32
   AUDIT ...................................................................................................................... 32

4.1 Architecture ......................................................................................................... 33
   4.1.1 Consent Management Component .............................................................. 33

5 References ............................................................................................................. 36

5.1 Patient Consent .................................................................................................... ¡Error! Marcador no definido.
5.2 Location ................................................................................................................ ¡Error! Marcador no definido.
5.3 Other ..................................................................................................................... ¡Error! Marcador no definido.
## Terms and abbreviations

<table>
<thead>
<tr>
<th>AB</th>
<th>Advisory Board</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>Consortium Agreement</td>
</tr>
<tr>
<td>DE</td>
<td>Decision Engine</td>
</tr>
<tr>
<td>NCP</td>
<td>National Contact Point</td>
</tr>
<tr>
<td>OLC</td>
<td>Open Location Code</td>
</tr>
<tr>
<td>PN</td>
<td>Participating Nation</td>
</tr>
</tbody>
</table>
Executive Summary

The objective of this deliverable is to set out the requirements for the design, development and implementation of consent management within the SHiELD project. This document provides background information surrounding consent management and also walks through each of the use cases defined in work package 6.1. This also details the integration points with other components of the SHiELD framework, such as data sensitivity tool, data hiding tool and the OpenNCP architecture.
1 Introduction

1.1 Objective and Scope

This work package is primarily concerned with extracting two main requirements from the three use cases defined in WP6. The first is the consent requirements and the second is how these consent preferences, along with the other requirements of each use case can be enforced during data exchange. The following sections will discuss each use case in this context and extract both the consent and the policy requirements.

1.2 Document Structure

Section 2 describes the overall principals of consent management along with the proposed data model and application programming interface (API) used to manage the capture and retrieval of consent.

Section 3 provides detailed analysis of the specific use cases defined in Deliverable D6.1 - use case specification [1] in relation to consent management.

Section 4 provides background information on the Symphonic product suite.
2 Consent Management

2.1 Overview

There is a requirement within the SHiELD project to capture and persist the citizen’s consent preferences for sharing data outside of their home country. These preferences will be set within the location of their home country and a mechanism will be created to enforce these preferences when transferring personal data beyond their home border. OpenNCP has been chosen as the cross-border information sharing framework. The Consent Manager component will expose a JSON HTTP API that allows clients to create, read update and delete consent preferences.

2.2 Data Model

The Consent Manager uses a deliberately simple data model, to enable us to clearly demonstrate the required functionality without first having to agree on a detailed specification on how patient consent should be represented. Each patient consent record consists of the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique Id for the consent (GUID).</td>
</tr>
<tr>
<td>patientId</td>
<td>Ideally this would be a globally unique ID for the patient.</td>
</tr>
<tr>
<td>documentId</td>
<td>Ideally this would be a globally unique ID e.g. an HL7 PID.3 Medical Record Number (MRN), (GUID) or could be a key to another table.</td>
</tr>
<tr>
<td>consent</td>
<td>Boolean value which is true if the patient has given consent, and false otherwise.</td>
</tr>
<tr>
<td>startDate</td>
<td>Date at which consent begins.</td>
</tr>
<tr>
<td>endDate</td>
<td>Date at which consent ends.</td>
</tr>
<tr>
<td>location</td>
<td>Geographic location in which consent is given. Possibly using OLC</td>
</tr>
<tr>
<td>purposeOfUse</td>
<td>Reason for performing one or more operations on information, which may be permitted by source system’s security policy in accordance with one or more privacy policies and consent directives.</td>
</tr>
</tbody>
</table>

2.3 API

The consent management tool will be accessible via a REST (Representational state transfer) interface. This API will be consumed by different user interfaces hosted within each country and realistically will reside in many different systems. For example, there may be a user interface within a GP practice software that enables the capture and modification of patient consent preferences. There may also be a citizen portal that allows citizens themselves to set their preferences. Although the production of a deployable UI is beyond the scope of this project it is anticipated, for demonstration purposes, that a simple UI will be produced to simulate described UI applications.
The exposed REST API definitions are described below in the following sections.

### 2.3.1 GET /api/consent (List all consents for a specific patient)

The server will produce an HTTP status 200 response, the body of which is a JSON document listing all consent records held for the patient, for example:

```json
[
  {
    "id": "c5829975-1b4b-4fc3-90a9-a2f5b959a479",
    "patientId": "b3705372-d284-4903-b959-5f676e6e46f9",
    "documentId": "bc547f01-ace6-4a2c-8818-faba793471c6",
    "consent": true
    "startDate": 2017-12-10
    "endDate": 2017-12-18
    "location": "6PH57VP3+PQ" (OLC)
  },
  {
    "id": "fbf6147a-72cb-4b46-a784-c40e98b0975a",
    "patientId": "b3705372-d284-4903-b959-5f676e6e46f9",
    "documentId": "af24af32-cb88-44b6-8d23-a8988580551d",
    "consent": true
    "startDate": 2017-12-10
    "endDate": 2017-12-18
    "location": "6PH57VP3+PQ" (OLC)
  },
...
```

### 2.3.2 GET /api/consent/{documentId} (Show consents for a specific document)

The server will produce an HTTP status 200 response, the body of which is a JSON document showing the consent records held for the given patient and document. If there is no consent record for the given ids, the server will produce an HTTP status 404 response. Example:
2.3.3 POST /api/consent  (Create a consent record)
{
    "patientId": "2b9f9655-6905-461d-9794-ac6aa73d32f8",
    "documentId": "1a1b23ca-de89-490b-a289-512d1aa31b58",
    "consent": true,
    "startDate": 2017-12-10,
    "endDate": 2017-12-18,
    "location": "6PH57VP3+PQ" (OLC)
}

If the body of the request is a valid JSON document of the form above, the Consent Manager will create a new consent record to match the request body. The server will then produce an HTTP status 200 response, the body of which will be identical to the body of the request message but with an added id. If the format of the request body is not valid, the server will produce an HTTP status 400 response.

2.3.4 PUT /api/consent/{id}  (Update a consent record)
{
    "patientId": "MRN12345^^^&2.16.840.1.113883.2.1.3.42.2.1.2.3.1.1.1.3.1&ISO",

"documentId": "2daa938b-9182-456a-b8fe-98aba041b778",
"consent": true
"startDate": 2017-12-10
"endDate": 2017-12-18
"location": "6PH57VP3+PQ" (OLC)
}

If the body of the request is a valid JSON document and the consented exists for the patient, the Consent Manager will update the state of the specified consent record to match the request body. The server will then produce an HTTP status 200 response, the body of which will be identical to the body of the request message. If the format of the request body is not valid, the server will produce an HTTP status 400 response.

2.3.5 DELETE /api/consent/{id} (Delete a consent record)
If a consent record with the specified id exists for the patient, it will be deleted, and the server will produce an HTTP status 200 response. If no such record exists, the server will produce an HTTP status 400 response.
3 Use Case Requirement Analysis

In this section we analyse the use cases data flow and approval needs. For each use case we provide the following: (1) a short summary and use case data flow. (2) We describe the sequence diagrams related to the data request and approval. (3) We provide the process that needs to be implemented in the enforcement point, and finally (4) we provide the outline for the necessary policies.

The following terms will be used when describing the inputs (attributes) that form the requests relating to the separate use cases.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>patientId</td>
<td>Ideally this would be a globally unique ID for the patient.</td>
</tr>
<tr>
<td>Hospital Id</td>
<td>Unique id for hospital</td>
</tr>
<tr>
<td>Clinician Id</td>
<td>Unique Id for clinician requesting informaiton</td>
</tr>
<tr>
<td>Originating PN</td>
<td>The participating nation unique id where the request originated from</td>
</tr>
<tr>
<td>Document Id</td>
<td>The unique id of the document being requested</td>
</tr>
<tr>
<td>Break Glass</td>
<td>A term used to indicate an emergency case where standard rules (policies) do not apply</td>
</tr>
<tr>
<td>Mobile Device Signature</td>
<td>A unique signature for a given mobile device</td>
</tr>
</tbody>
</table>

3.1 Use Case 1 – ‘Break Glass’ Scenario

The medical clinician can enter a ‘Break Glass’ mode which would allow her access to additional data. As part of WP2, the developed architecture would need to support the ‘Break Glass’ mode and ensuring this is passed along with the OpenNCP request by way of metadata. There may already be a mechanism within OpenNCP to add this additional metadata to the request (requires further investigation).

Example Request:

```xml
<Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:purposeofuse" ...>
  <AttributeValue>BREAK_GLASS</AttributeValue>
</Attribute>
```

3.1.1 Key components Diagram

Figure 1 below shows the key components involved in the data flow within the system. Note that all data passes through enforcement point(s), these points consult the decision engine for a decision as to whether to allow or prevent the data from continuing.
3.1.2 Sequence Diagram - Document Listing

Use Case 1 - Document Listing Request Sequence

Figure 1: Use Case 1 - Component Overview Diagram
Figure 2: Use Case 1 - Sequence Diagram - Document Listing

3.1.3 Sequence Diagram - Document Viewing

Use Case 1 - Document Viewing Request Sequence

Figure 3: Use Case 1 – Sequence Diagram – Document Viewing
3.1.4 Enforcement Point

Next, we expand on the processes that taking place in each step:

- **Document Listing Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Local Authorization Decision Request’ before proceeding.
  - Make a ‘Remote Document Listing Request’ to NCP Italy.
  - Return the list of documents from the previous request.

- **Local Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
  - Actions:
    - Authorization
  - Domains:
    - Local

- **Remote Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
    - Originating PN
  - Actions:
    - Authorization
  - Domains:
    - Remote

- **Remote Document Listing Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Remote Authorization Decision Request’ before proceeding.
  - Get document listing for patient from local EHR and any other local sources.
  - For each document make a request to Decision Engine:
    - Attributes:
      - Patient ID
      - Document ID
      - ‘Break Glass’
    - Actions:
      - Read.Listing - Same as Read.Document but without redaction obligations.
  - Return a list of all documents which received a PERMIT.

- **View Document Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Local Authorization Decision Request’ before proceeding.
  - Make a ‘Remote View Document Request’ against the relevant NCP.

- **Remote View Document Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
3.1.5 Decision Engine Policies

- **Policy Set to check local request is authorized:**
  - **Combining Algorithm:** Permit unless deny.
  - **Domain:** Local.
  - **Action:** Authorization.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
  - Policy to deny unless clinician has the correct level of access.

- **Policy Set to check remote request is authorized:**
  - **Combining Algorithm:** Permit unless deny.
  - **Domain:** Remote.
  - **Action:** Authorization.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
  - Policy to deny unless clinician has the correct level of access.
  - Policy to deny unless requesting PN is on approved list.

- **Policy Set to view a document:**
  - **Combining Algorithm:** Deny unless permit
  - **Domain:** Local
  - **Action:** Read.Document
  - Policy to permit if ‘Break Glass’ is true and document is on the ‘Break Glass’ list.
    - Obligation applies to permit, containing redaction instructions. This will contain an attribute which resolves against the data sensitivity service.
  - Policy to permit if the patient has given consent for this document.
    - Rule to permit if global consent is given.
    - Obligation applies to permit, containing redaction instructions. This will contain an agreed coding which can be used by the ‘Enforcement Point’ to call the necessary data hiding service (delivered as part of WP5.4).
3.2 Use Case 2: Surgical Intervention

This use case demonstrates the ability to update patient consent data via a mobile device. Retrieval of remote data from another PN is also covered. In addition, this scenario showcases the ability to enact more complex privacy policies dependant on time and location. The policies described in the following section build upon those described in Use Case one such that they handle both use cases one and two.

3.2.1 Key components Diagram

![Diagram of Use Case 2 - Key components](image)

*Figure 4: Use Case 2 - Key components Diagram*
3.2.2 Sequence Diagram - Consent Update via Mobile Device

Use Case 2 - Update Patient Consent via Mobile Device

Figure 5: Use case 2 - Update Patient Consent via Mobile Device Sequence Diagram
3.2.3 Sequence Diagram - Document Listing

Figure 6: Use Case 2 - EHR Data Retrieval Sequence Diagram
3.2.4 Sequence Diagram - Document Viewing

Use Case 2 - Document Viewing Request Sequence

![Sequence Diagram](image)

Figure 7: Use Case 2 - EHR Data Retrieval Sequence Diagram
3.2.5 Enforcement Point

Additional functionality required to facilitate this use case is highlighted in green.

- **Document Listing Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Local Authorization Decision Request’ before proceeding.
  - Make a ‘Remote Document Listing Request’ to NCP Italy.
  - Return the list of documents from the previous request.

- **Local Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
  - Actions:
    - Authorization
  - Domains:
    - Local

- **Remote Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
    - Originating PN
  - Actions:
    - Authorization
  - Domains:
    - Remote

- **Remote Document Listing Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Remote Authorization Decision Request’ before proceeding.
  - Get document listing for patient from local EHR and any other local sources.
  - For each document make a request to Decision Engine:
    - Attributes:
      - Patient ID
      - Document ID
      - Request Origin (Location)
      - ‘Break Glass’
    - Actions:
      - Read.Listing - Same as Read.Document but without redaction obligations.
  - Return a list of all documents which received a PERMIT.

- **View Document Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Local Authorization Decision Request’ before proceeding.
  - Make a ‘Remote View Document Request’ against the relevant NCP.

- **Remote View Document Request:**
- Ensure the request is Authenticated using openNCP before proceeding.
- Make a ‘Remote Authorization Decision Request’ before proceeding.
- Make a decision request to Decision Engine:
  - Attributes:
    - Patient ID
    - Document ID
    - Request Origin (Location)
    - 'Break Glass'
  - Actions:
    - Read.Document
  - Domains:
    - Local
- Enforce any redaction obligations returned from the Decision Engine.
- Return the redacted document.

- **Update Consent via Mobile Interface:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Mobile Device Authorization Decision Request’ before proceeding.
  - Make updates to consent DB.
  - Return success / failure response to mobile device.

- **Mobile Device Authorization Decision Request:**
  - Attributes:
    - Patient ID
    - Mobile Device Signature
  - Actions:
    - Authorization
  - Domains:
    - MobileDevice

### 3.2.6 Decision Engine Policies

Additional functionality required to facilitate this use case is highlighted in green.

- **Policy Set to check a mobile device is authorized:**
  - **Combining Algorithm:** Permit unless deny.
  - **Domain:** MobileDevice.
  - **Action:** Authorization.
  - Policy to check Patient ID is correct.
  - Policy to check mobile device signature is correct.
  - Policy to check patient and device are registered with this PN.

- **Policy Set to check local request is authorized:**
  - **Combining Algorithm:** Permit unless deny.
  - **Domain:** Local.
  - **Action:** Authorization.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
Policy to deny unless clinician has the correct level of access.

- **Policy Set to check remote request is authorized:**
  - Combining Algorithm: Permit unless deny.
  - Domain: Remote.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
  - Policy to deny unless clinician has the correct level of access.
  - Policy to deny unless requesting PN is on approved list.

- **Policy Set to view a document:**
  - Combining Algorithm: Deny unless permit
  - Domain: Local
  - Action: Read.Document
  - Policy to permit if ‘Break Glass’ is true and document is on the ‘Break Glass’ list.
    - Obligation applies to permit, containing redaction instructions. This will contain an attribute which resolves against the data sensitivity service.
  - Policy to permit if the patient has given consent for this document.
    - Combining Algorithm: ‘Condition 1’ or (‘Condition 2’ and ‘Condition 3’)
      - Condition 1 to permit if global consent is given.
      - Condition 2 to permit if consent is given for location from which the request originated.
      - Condition 3 to permit if consent is within specified time frame.
      - Obligation applies to permit, containing redaction instructions. This will contain an attribute which resolves against the data sensitivity service.
3.3 Use Case 3: Chronic Conditions + Remote Monitoring

Use case 3 describes a patient, travelling abroad with chronic conditions, and wishes to consent to have their EHR checked to allow medical staff in other countries operating under the ShiELD platform to access her patient summary record. While abroad, the patient experiences symptoms necessitating a visit to a hospital in the country to which they are travelling. When the summary is accessed, the medical staff in the hospital may identify that clinical data outside of that held in the patient’s patient summary record may be useful in understanding and analysing the patient’s condition.

The policies described in the following section build upon those described previous use cases such that they handle all three use cases.

3.3.1 Data Flow Diagram

![Data Flow Diagram](image-url)

*Figure 8: Use Case 3 - Data Flow Diagram*
3.3.2 Sequence Diagram – Document Listing Multiple Jurisdictions

Figure 9: Use Case 3 – Document Listing Multiple Jurisdictions
3.3.3 Sequence Diagram - Document Viewing

Use Case 3 - Document Viewing Request Sequence

![Sequence Diagram]

Figure 10: Use Case 3 - EHR Retrieval
3.3.4 Enforcement Point

Additional functionality required to facilitate this use case is highlighted in green.

- **Global Document Listing Request:**
  - Perform the steps in a ‘Local Document Listing Request’ for the current PN
    - This will use ‘Local Authorization Decision Request’
  - Make a ‘Local Document Listing Request’ to all PNs.
    - This will use ‘Remote Authorization Decision Request’
  - Return a collated list of documents retrieved from the previous steps.

- **Local Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
  - Actions:
    - Authorization
  - Domains:
    - Local

- **Remote Authorization Decision Request:**
  - Attributes:
    - Hospital ID
    - Clinician ID / Role
    - Originating PN
  - Actions:
    - Authorization
  - Domains:
    - Remote

- **Local Document Listing Request:**
  - Ensure the request is Authenticated using openNCP before proceeding.
  - Make a ‘Remote Authorization Decision Request’ or ‘Local Authorization Decision Request’ as appropriate before proceeding.
  - Get document listing for patient from local EHR and any other local sources.
    - This will include references to external data within the PN such as “Carpeta Salud” app data.
    - The links to external data will need to be provided by a PIP within the NCP.
    - Patient consent for external data sources will need to be stored here, (the patient consent service could also act as the PIP mentioned in the previous point).
  - For each document make a request to Decision Engine:
    - Attributes:
      - Patient ID
      - Document ID
      - Request Origin (Location)
      - ‘Break Glass’
    - Actions:
• **Read.Listing** - Same as Read.Document but without redaction obligations.
  o Return a list of all documents which received a PERMIT.

• **View Document Request:**
  o Ensure the request is Authenticated using openNCP before proceeding.
  o Make a **Local Authorization Decision Request** before proceeding.
  o Make a Local or Remote **View Document Request** as required.

• **Remote View Document Request:**
  o Ensure the request is Authenticated using openNCP before proceeding.
  o Make a **Remote Authorization Decision Request** before proceeding.
  o Make a decision request to Decision Engine:
    - Attributes:
      o Patient ID
      o Document ID
      o Request Origin (Location)
      o 'Break Glass'
    - Actions:
      o Read.Document
  o Retrieve the requested document.
    - This could be from EHR or an external service such as “Carpeta Salud” app data.
  o Enforce any redaction obligations returned from the Decision Engine.
  o Return the redacted document.

• **Update Consent via Mobile Interface:**
  o Ensure the request is Authenticated using openNCP before proceeding.
  o Make a ‘Mobile Device Authorization Decision Request’ before proceeding.
  o Make updates to consent DB.
  o Return success / failure response to mobile device.

• **Mobile Device Authorization Decision Request:**
  o Attributes:
    - Patient ID
    - Mobile Device Signature
  o Actions:
    - Authorization
  o Domains:
    - MobileDevice
3.3.5 Decision Engine Policies

Additional functionality required to facilitate this use case is highlighted in green.

- **Policy Set to check a mobile device is authorized:**
  - Combining Algorithm: Permit unless deny.
  - Domain: MobileDevice.
  - Policy to check Patient ID is correct.
  - Policy to check mobile device signature is correct.
  - Policy to check patient and device are registered with this PN.

- **Policy Set to check local request is authorized:**
  - Combining Algorithm: Permit unless deny.
  - Domain: Local.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
  - Policy to deny unless clinician has the correct level of access.

- **Policy Set to check remote request is authorized:**
  - Combining Algorithm: Permit unless deny.
  - Domain: Remote.
  - Policy to deny unless the authentication credentials are correct by checking with openNCP authentication component.
  - Policy to deny unless requesting hospital is on an approved list.
  - Policy to deny unless clinician has the correct level of access.
  - Policy to deny unless requesting PN is on approved list.

- **Policy Set to view a document:**
  - Combining Algorithm: Deny unless permit
  - Domain: Local
  - Action: Read.Document
  - Policy to permit if ‘Break Glass’ is true and document is on the ‘Break Glass’ list.
    - Obligation applies to permit, containing redaction instructions. This will contain an attribute which resolves against the data sensitivity service.
  - Policy to permit if the patient has given consent for this document.
    - Combining Algorithm: ‘Condition 1’ or (‘Condition 2’ and ‘Condition 3’)
      - Condition 1 to permit if global consent is given.
      - Condition 2 to permit if consent is given for location from which the request originated.
      - Condition 3 to permit if consent is within specified time frame.
      - Obligation applies to permit, containing redaction instructions. This will contain an attribute which resolves against the data sensitivity service.
4 Consent Management Tools requirements

Symphonic suite provides an integrated set of tools that supports and enables the creation of a formal structure for abstraction, governance and implementation of trust relationships and security policies. Working across multiple disparate organisations and technologies, it provides a standardised trusted mechanism between all parties for sharing data, whilst maintaining strict conformance to the strongly defined trust framework.

The system is comprised of a number of core components, each of which can operate as stand-alone products or can integrate with existing systems. These are:

**TRUST**

- Define Trust Framework
  - Allows the definition of the components of the Trust Framework including Users, Roles, Organisations, Identity Providers, Attribute Providers, Trust Levels, Relationships, etc.
- Build Policy Rules
  - Creates strongly typed policies for access to data or services, tightly coupled to the trust framework
  - Enhances policies with highly customisable logic covering elements such as date/time, geo-location, and legitimate interest
  - Queries, reports and provides visualisation of the complete trust eco-system.

**GOVERNANCE**

- Validate Policy Rules
  - Resolves issues with rule sets, including overlaps, shadows, anomalies and conflicts
- Optimise Rules
  - Patent pending technology optimises analysis and decision processes producing a highly efficient decision table
- Decision Engine
  - Highly scalable and distributable decision engine supports advanced attribute retrieval and request processing.

**GATEWAY**

- Real-time policy enforcement
  - Gathers claims from users’ access requests and resolve required attributes to support the decision engine – ultimately providing a permit or deny along with explanatory reasons and advice.
- Trusted data access gateway
  - A portal/proxy to the real source of data, acts as the trusted gate-keeper to the data, permitting access only to requests which conform to the pre-defined policies and trust framework.

**AUDIT**

Underpinning these core components is a comprehensive auditing tool giving access to fine-grained audit and logging features.
• Audit Policy Management
  o Creates a detailed audit trail of all trust framework and policy changes.
• Audit Access Rights
  o Allows exploration of access rights granted/denied by the policies.
• Audit Access Requests
  o Creates an audit log of all user access requests made through the governance engine.

These components can be deployed and used as a full end-to-end solution for policy abstraction, implementation and controlled access to services and data, or can integrate each of the elements “as a Service” to existing applications.

These set of tools will be used within the SHiELD project to help define and building decision policies that will provide the basis for authorisation inter-country data requests as specified in the different use cases detailed in work package 6.1.

4.1 Architecture

The diagrams below provide an overview of the logical components that will make up our proposed solution. The consent management API component provides the main interface between the patient consent client and the core policy management and decision engine.

4.1.1 Consent Management Component

Figure 11: Consent Management Component Data Flow Diagram
Figure 12: Consent Management Component Detail
Figure 13: Decision Point Architecture
5 References


