

WS MyCareNet eAgreement

Cookbook
Version 1.0

This document is provided to you free of charge by the

eHealth platform

Willebroekkaai 38 – 1000 Brussel
38, Quai de Willebroeck – 1000 Bruxelles

All are free to circulate this document with reference to the URL source.

Table of contents

Table of contents	2
1 Document management	4
1.1 Document history	4
2 Introduction.....	5
2.1 Goal of the service.....	5
2.2 Goal of the document.....	5
2.3 eHealth document references.....	5
2.4 External document references	6
3 Business and privacy requirements.....	7
3.1 For issues in production	7
3.2 For issues in acceptance	7
3.3 For business issues	7
3.4 Certificates.....	7
3.5 Support desk CIN/NIC - contact points.....	7
3.5.1 eAgreement business support	7
3.5.2 MyCareNet Helpdesk:	7
3.5.3 Technical contact center MyCareNet:.....	7
4 Global overview.....	8
5 Step-by-step	9
5.1 Technical requirements	9
5.1.1 Use of the eHealth SSO solution	9
5.1.2 Encryption	9
5.1.3 Security policies to apply.....	10
5.2 Therapeutic link verification	10
5.3 Web service	10
5.3.1 Method AskAgreement	11
5.3.2 Method ConsultAgreement	14
5.3.3 Used Types	17
6 Security.....	18
6.1 Business security	18
6.2 Web service	18
7 Test and release procedure.....	19
7.1 Procedure	19
7.1.1 Initiation	19
7.1.2 Development and test procedure	19
7.1.3 Release procedure	19
7.1.4 Operational follow-up	19
7.2 Test cases.....	19
8 Error and failure messages.....	20



To the attention of: "IT expert" willing to integrate this web service.



1 Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
1.0	06/08/2020	eHealth platform	Initial version

2 Introduction

2.1 Goal of the service

Some treatments require a medical agreement in order to be able to claim reimbursement from the insurance organizations (IO). The "eAgreement" service will allow an authorized healthcare provider to send these requests for medical agreement electronically and also to consult them. The goal is to come up with a global solution so that this service can be reused for any type of agreement. The healthcare provider needs to request a SAML token from the eHealth Secure Token Service (STS) prior to calling the eAgreement services.

2.2 Goal of the document

This document is not a development or a programming guide for internal applications. Instead, it provides functional and technical information and allows an organisation to integrate and use the eHealth service.

However, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth partners must commit to comply with the requirements of specifications, data format and release processes described in this document.

Technical and business requirements must be met in order to allow the integration and validation of the eHealth service in the client application.

Detailed description of the functionality of the service, the semantics of the particular elements and other general information about the service is out of the scope of this document. This kind of information can be found in the documentation provided by MyCareNet on their Sharepoint.

In order to be able to test the MyCareNet eAgreement service, you need to take the following steps (see also section 5):

1. **Create a test case:** If the testing is done for a real care provider, the real NIHI number of the care provider can be used. Otherwise, you will receive a test NIHI number from the eHealth development team (you must indicate the service called and the kind of profile needed). You always need to request the configuration of the test cases at eHealth.
2. **Request an eHealth test certificate:** a test certificate must be requested at the eHealth platform.
3. **Obtain the SAML token from the STS:** the eHealth test certificate obtained in the previous step is used for identification at the STS and as the Holder-Of-Key (HOK) certificate.
4. **Call the eAgreement web services.**

2.3 eHealth document references

All the document references can be found on the eHealth portal¹. These versions or any following versions can be used for the eHealth service.

ID	Title	Version	Date	Author
1	Glossary.pdf	1.0	01/01/2010	eHealth platform
2	Cookbook STS	1.2	13/04/2018	eHealth platform
3	eAgreement_SSO.pdf	1.0	06/08/2020	eHealth platform

¹ <https://www.ehealth.fgov.be/ehealthplatform>



2.4 External document references

All the MyCareNet documentation can be found within their Sharepoint². The documentation referenced in this section may evolve in time.

If some external documentation has been modified, please notify the eHealth service management³ who manages the maintenance of this document.

ID	Title	Last modification date	Author
1.	Web Service Security – SAML Token profile 1.1 http://www.oasis-open.org/committees/download.php/16768/wssv1.1-spec-os-SAMLSecurityProfile.pdf	01/02/2006	OASIS
2.	CIN Messages definition eAgreement v1	05/05/2020	CIN
4.	GenericSync Error codes	13/05/2020	CIN
5.	MyCareNet Authentication Catalogue	05/05/2020	CIN
6.	ImplementationGuide_For_CareProvider	05/05/2020	CIN
7.	NIPPIN GenSync V3 (ESB 2 NIPPIN)	05/05/2020	CIN
9.	Service_Catalogue_iSocial_Commons	05/05/2020	CIN
10.	Service_Catalogue_iSocial_GenSync	05/05/2020	CIN
11.	xsd-encryption	05/05/2020	CIN
12.	3-EN-DATA-KINE-ALL_eAgreement-Messages definition	03/08/2020	CIN
13.	Draft FHIR messages 20200731	31/07/2020	CIN
14.	Implementation Guide 20200731	31/07/2020	CIN

² In order to have access to the Sharepoint, you need to create an account which can be requested at : <http://fra.mycarenet.be/wie-zijn-we/contact> or <http://ned.mycarenet.be/wie-zijn-we/contact>

³ ehealth_service_management@ehealth.fgov.be

3 Business and privacy requirements

3.1 For issues in production

eHealth platform contact center:

- Phone: 02/788 51 55
- Mail: support@ehealth.fgov.be
- Contact Form :
 - <https://www.ehealth.fgov.be/ehealthplatform/nl/contact> (Dutch)
 - <https://www.ehealth.fgov.be/ehealthplatform/fr/contact> (French)

3.2 For issues in acceptance

Integration-support@ehealth.fgov.be

3.3 For business issues

- regarding an existing project: the project manager in charge of the application or service
- regarding a new project and other business issues: info@ehealth.fgov.be

3.4 Certificates

- In order to access the secured eHealth platform environment you have to obtain an eHealth platform certificate, used to identify the initiator of the request. In case you do not have one please consult the chapter about the eHealth Certificates on the portal of the eHealth platform

<https://www.ehealth.fgov.be/ehealthplatform/nl/ehealth-certificaten>

<https://www.ehealth.fgov.be/ehealthplatform/fr/certificats-ehealth>

- For technical issues regarding eHealth platform certificates

Acceptance: acceptance-certificates@ehealth.fgov.be

Production: support@ehealth.fgov.be

3.5 Support desk CIN/NIC - contact points

3.5.1 eAgreement business support

For business questions related to eAgreement: MyCareNet Helpdesk (first line support)

3.5.2 MyCareNet Helpdesk:

Telephone: 02/891 72 00

Mail: support@intermut.be

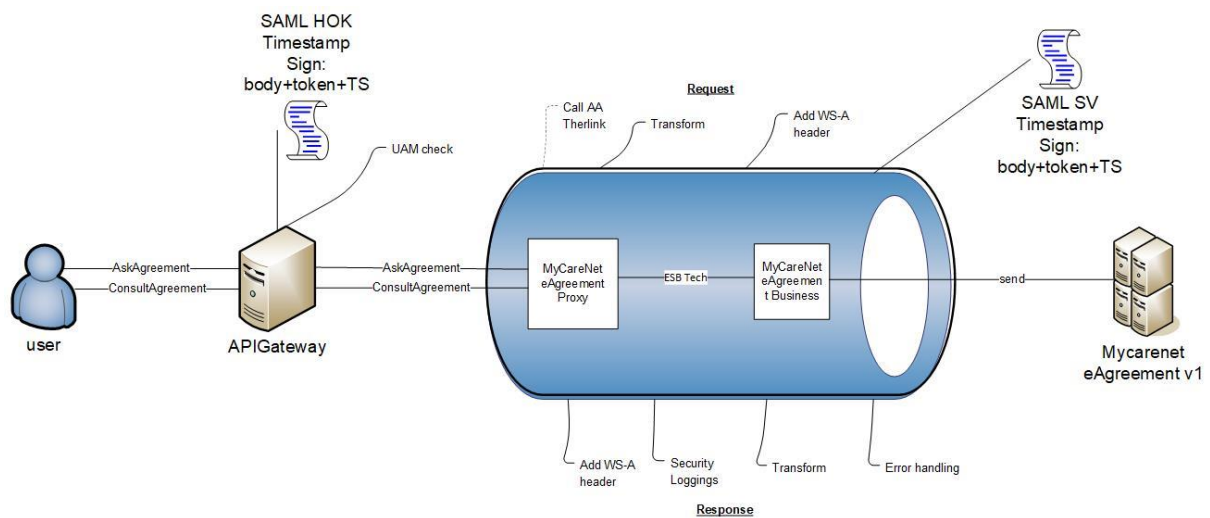
3.5.3 Technical contact center MyCareNet:

Telephone: 02/431 47 71

Mail: ServiceDesk@MyCareNet.be



4 Global overview



The eAgreement WS is secured with the SAML HOK policy. Therefore, prior to calling the services, a SAML token must be obtained at the eHealth STS. The obtained token must be then included in the header of the request message, where the timestamp and the body must be signed with the certificate as used in the HOK profile of the SAML token (more detailed technical description can be found further in the chapter 5 of this cookbook). The body contains the eAgreement request (Ask, or Consult). The eHealth ESB verifies the security (authentication, authorization, etc.) and forwards the request to MyCareNet. Then, the service returns the response delivered by the MyCareNet backend.

Note:

If the operation is "ConsultAgreement" AND the user is a professional physiotherapist, the eHealth ESB executes a call out to AttributeAuthority service in order to verify the therapeutic link between the physiotherapist and the patient. This is described in more details in section 5.2.

5 Step-by-step

5.1 Technical requirements

In order to test the service, the eHealth development team first has to create a test case. The rules to access the eAgreement WS are the same in acceptance and in production.

Access rules:

- authentication with a care providers certificate;
- authentication with the certificate of a mandate holder.

The eHealth development team has to configure all test cases.

So, before doing any test, request your test cases from the eHealth development team (info@ehealth.fgov.be).

In order to implement a WS call protected with a SAML token you can reuse the implementation as provided in the "eHealth technical connector". Nevertheless, eHealth implementations use standards and any other compatible technology (WS stack for the client implementation) can be used instead.

- <https://www.ehealth.fgov.be/ehealthplatform/nl/service-ehealth-platform-services-connectors>
- <https://www.ehealth.fgov.be/ehealthplatform/fr/service-ehealth-platform-services-connectors>

Alternatively, you can write your own implementation. The usage of the STS and the structure of the exchanged xml-messages are described in the eHealth STS cookbook.

- <https://www.ehealth.fgov.be/ehealthplatform/fr/service-iam-identity-access-management>
- <https://www.ehealth.fgov.be/ehealthplatform/nl/service-iam-identity-access-management>

5.1.1 Use of the eHealth SSO solution

This section specifies how to call the STS in order to have access to the WS. You must precise several attributes in the request. The details on the identification attributes and the certification attributes can be found in the separate document eAgreement_SSO.

To access the eAgreement WS, the response token must contain "true" for all of the 'boolean' certification attributes and a non-empty value for other certification attributes.

If you obtain "false" or empty values, contact the eHealth platform to verify that they correctly configured the requested test case.

5.1.2 Encryption

All the information about the use of the encryption libraries and the call to the eHealth Token Key (ETK) depot are described in the End-To-End Encryption (ETEE) cookbooks on the portal of the eHealth platform.

To encrypt the request parts, you have to call the GetEtk operation to pick up the right ETK from the eHealth ETK depot. For example, the table below provides you the identifiers to use in the GetEtkRequest.

Environment	Type	Value	Application ID
Integration Test Environment	CBE	0820563481	MYCARENET
Acceptance Environment	CBE	0820563481	MYCARENET
Production Environment	CBE	0820563481	MYCARENET



5.1.3 Security policies to apply

We expect that you use SSL one way for the transport layer.

To call the eAgreement WS:

- Add the business message to the soap body
- Add to the SOAP header the following elements:
 - **SAML Token**: The SAML assertion received from the eHealth STS. This assertion needs to be forwarded exactly as received in order to not to break the signature of the eHealth STS. The token needs to be added accordingly to the specifications of the OASIS SAML Token Profile (HOK)).
 - **Timestamp**
 - A **signature** that has been placed on the SOAPBody and the timestamp with the certificate of which the public key is mentioned in the SAML Assertion.
- The signature element (mentioned above) needs to contain:
 - SignedInfo with References to the soapBody and the Timestamp.
 - KeyInfo with a SecurityTokenReference pointing to the SAML Assertion.

See also the WSSP in the WSDL⁴ (also included in the documentation).

5.2 Therapeutic link verification

As explained previously, a physiotherapist can consult the agreement of a patient if and only if it exists a therapeutic link between them.

This verification is made in the ESB eHealth.

If the therapeutic link doesn't exist, the physiotherapist receives an error SOA-01002 (see section 8 for more details about the SOA errors)

5.3 Web service

The eAgreement WS has two operations available:

- AskAgreement
- ConsultAgreement

The eAgreement WS has the following endpoints:

- Pilot environment: <https://services-acpt.ehealth.fgov.be/MyCareNet/eAgreement/v1>
- Production environment: **<https://services.ehealth.fgov.be/MyCareNet/eAgreement/v1>**

The remainder of this section describes the structure of the request and the response messages. For more details on the specific elements and the concepts behind them, see the documentation as provided by the CIN/NIC on their Sharepoint.

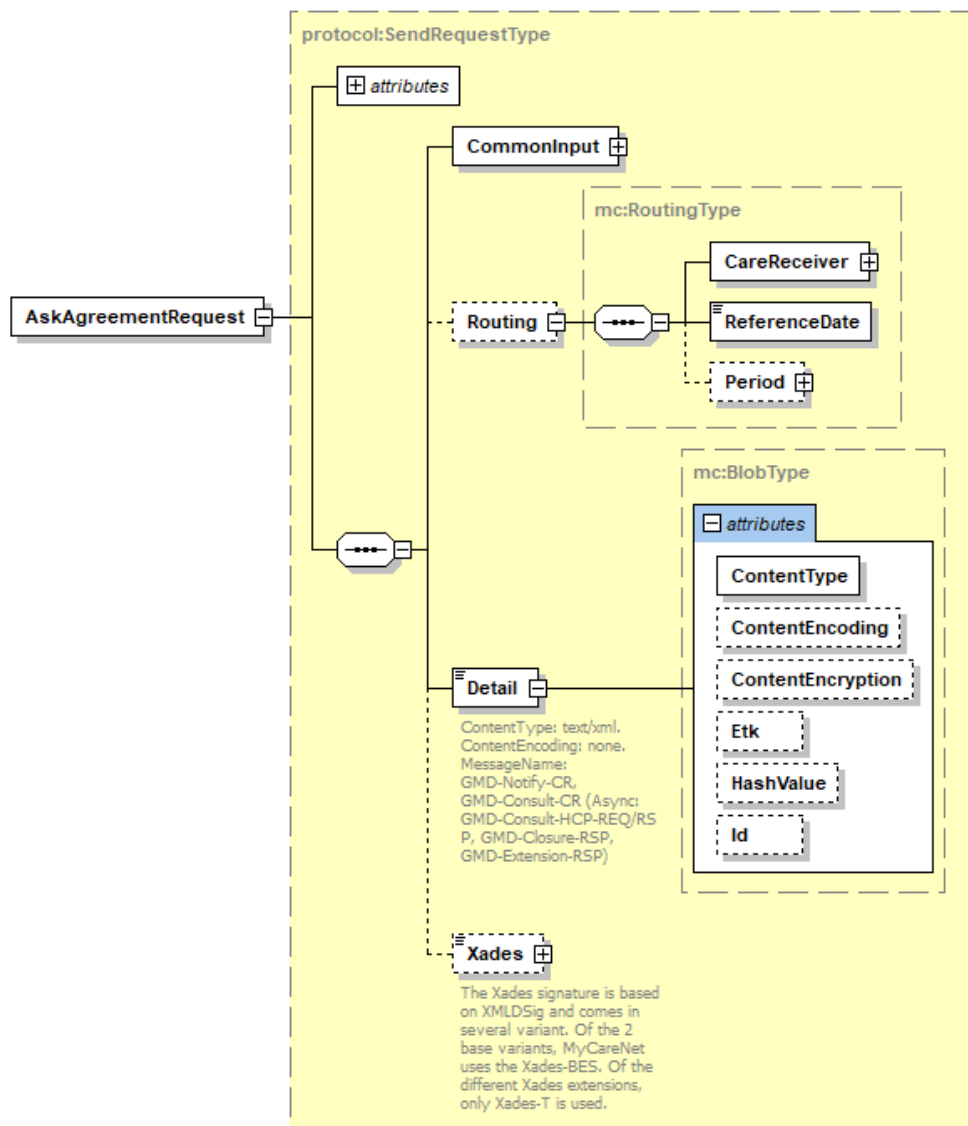
⁴ WSDL's can be found in the eHealth Service Registry: <https://services.ehealth.fgov.be/registry/uddi/bsc/web>

5.3.1 Method AskAgreement

This method allows

- to request a new agreement to the insurance institutions OR
- to argue a request being processed OR
- to extend an existing agreement (paper or electronic) OR
- to cancel a request in the event of an error (as long as it is not already processed)

5.3.1.1 Input arguments in AskAgreementRequest



Field name	Description
CommonInput	See section 5.3.6.1 : CommonInputType
Routing	Mandatory element. See the documentation 'Service_Catalogue_iSocial_Common' provided by the CIN/NIC. The data within this element should contain the SSIN of the care receiver.

Detail	<p>Encrypted detail of the request. The content of the encrypted message should respect some standard format to allow additional information exchange:</p> <ul style="list-style-type: none"> - The identity of the Key to be used to encrypt the response. - The XAdES as probative force of the message. <p>See the documentation provided by the CIN/NIC for more details about the structure "EncryptedKnownContent":</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync' <p>Attribute values :</p> <p>@ContentType: "text/xml"</p> <p>@ContentEncoding: "none"</p> <p>@ContentEncryption: "encryptedForKnownBED"</p> <p>@ETK the encryption token that has been used for the encryption of the body. Mentioning this information helps the recipient to identify the private key to be used for decryption. When not provided, the recipient may choose to reject the message or try to decrypt using the several existing private keys.</p> <p>@HashValue pre-calculated hash of the uncompressed and decoded content. Is always provided to the care provider.</p> <p>@Id: The ID of the blob for usage in the XAdES signature. It is an "NCName" instead of an "ID" in order to be able to have different blobs with the same (fixed) id without causing an XSD validation.</p> <p>Note that the attribute "MessageName" in the Detail element is not present in the interface as provided by eHealth. This attribute value is then filled out by the eHealth platform according to the called operation (for the AskAgreement message it is "eAgreement-ask").</p>
Xades	<p>The Xades must be inserted in the "EncryptedKnownContent" structure.</p> <p>See the documentation provided by the CIN/NIC for more details about the structure "EncryptedKnownContent":</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync'

5.3.1.2 Request example

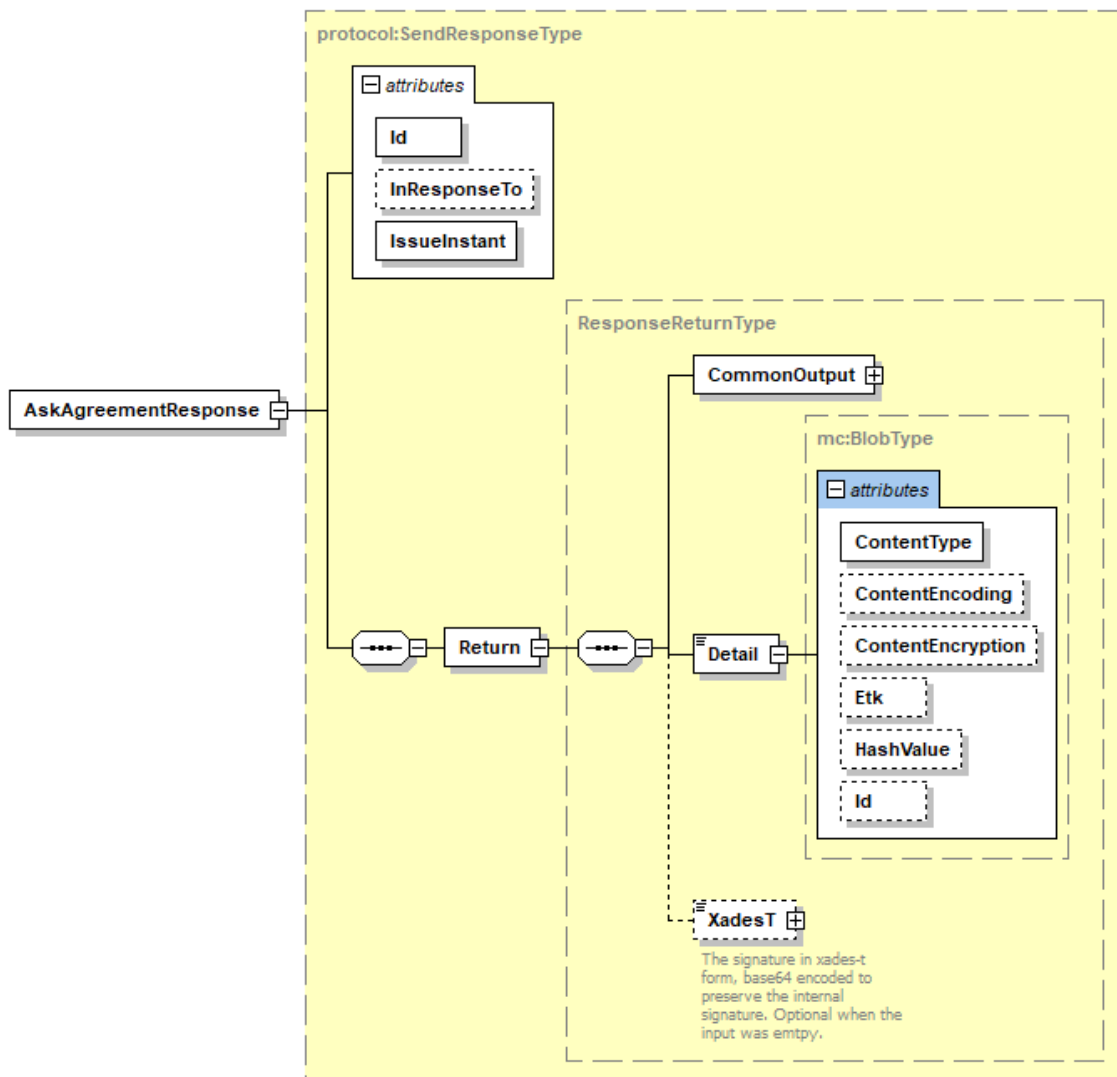
Business example is generated from the documentation

- 'Draft FHIR messages 20200731'
- 'Implementation Guide 20200731'

provided by CIN/NIC.



5.3.1.3 Output arguments in AskAgreementResponse



Field name	Description
"Response"	<p>@Id: Unique Id for tracing</p> <p>@InResponseTo: 'Id' attribute of the request if available</p> <p>@IssueInstant: Generation response moment</p>
Return	<p>See the documentation provided by the CIN/NIC for more details :</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync' - 'Draft FHIR messages 20200731' - 'Implementation Guide 20200731' <p>NB: In this case, the attribute <code>@ContentEncryption</code> can only have the value "encryptedForKnownRecipient" (the content of the body is encrypted with the public key of the health-care provider).</p>

5.3.1.4 Response example

Business example is generated from the documentation

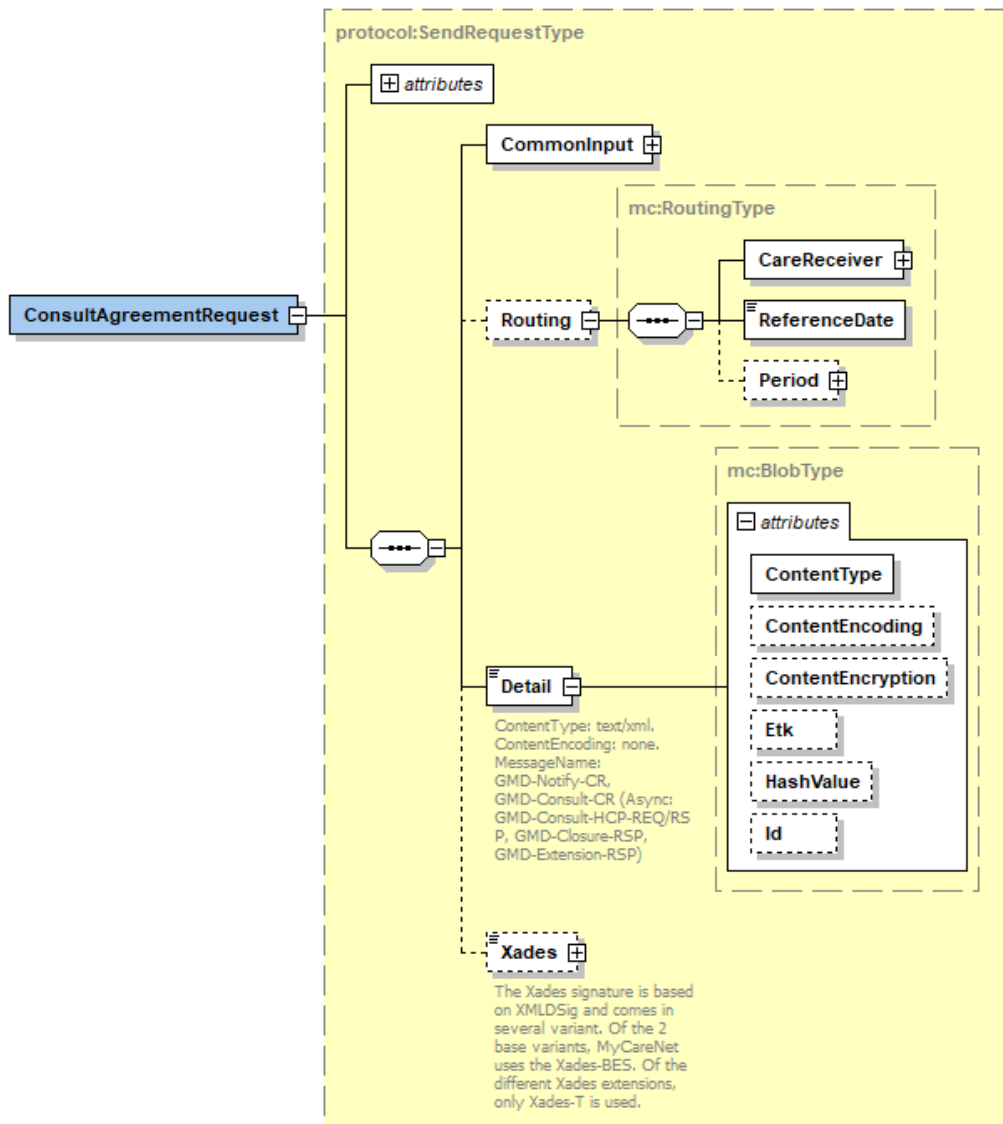


- 'Draft FHIR messages 20200731'
 - 'Implementation Guide 20200731'
- provided by CIN/NIC.

5.3.2 Method ConsultAgreement

The goal of this method is to consult the requests for agreement from a patient.

5.3.2.1 Input arguments in ConsultAgreementRequest



Field name	Description
CommonInput	See section 5.3.6.1 CommonInputType
Routing	Mandatory element. See the documentation 'Service_Catalogue_iSocial_Common' provided by the CIN/NIC. The data within this element should contain the SSIN of the care receiver.

Detail	<p>Encrypted detail of the request. The content of the encrypted message should respect some standard format to allow additional information exchange:</p> <ul style="list-style-type: none"> - The identity of the Key to be used to encrypt the response. - The XAdES as probative force of the message. <p>See the documentation provided by the CIN/NIC for more details about the structure "EncryptedKnownContent":</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync' <p>Attribute values :</p> <p>@ContentType: "text/xml"</p> <p>@ContentEncoding: "none"</p> <p>@ContentEncryption: "encryptedForKnownBED"</p> <p>@ETK the encryption token that has been used for the encryption of the body. Mentioning this information helps the recipient to identify the private key to be used for decryption. When not provided, the recipient may choose to reject the message or try to decrypt using the several existing private keys.</p> <p>@HashValue pre-calculated hash of the uncompressed and decoded content. Is always provided to the care provider.</p> <p>@Id: The ID of the blob for usage in the XAdES signature. It is an "NCName" instead of an "ID" in order to be able to have different blobs with the same (fixed) id without causing an XSD validation.</p> <p>Note that the attribute "MessageName" in the Detail element is not present in the interface as provided by eHealth. This attribute value is then filled out by the eHealth platform according to the called operation (for the ConsultAgreement message it is "eAgreement-consult").</p>
Xades	<p>The Xades must be inserted in the "EncryptedKnownContent" structure.</p> <p>See the documentation provided by the CIN/NIC for more details about the structure "EncryptedKnownContent":</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync'

5.3.2.2 Request example

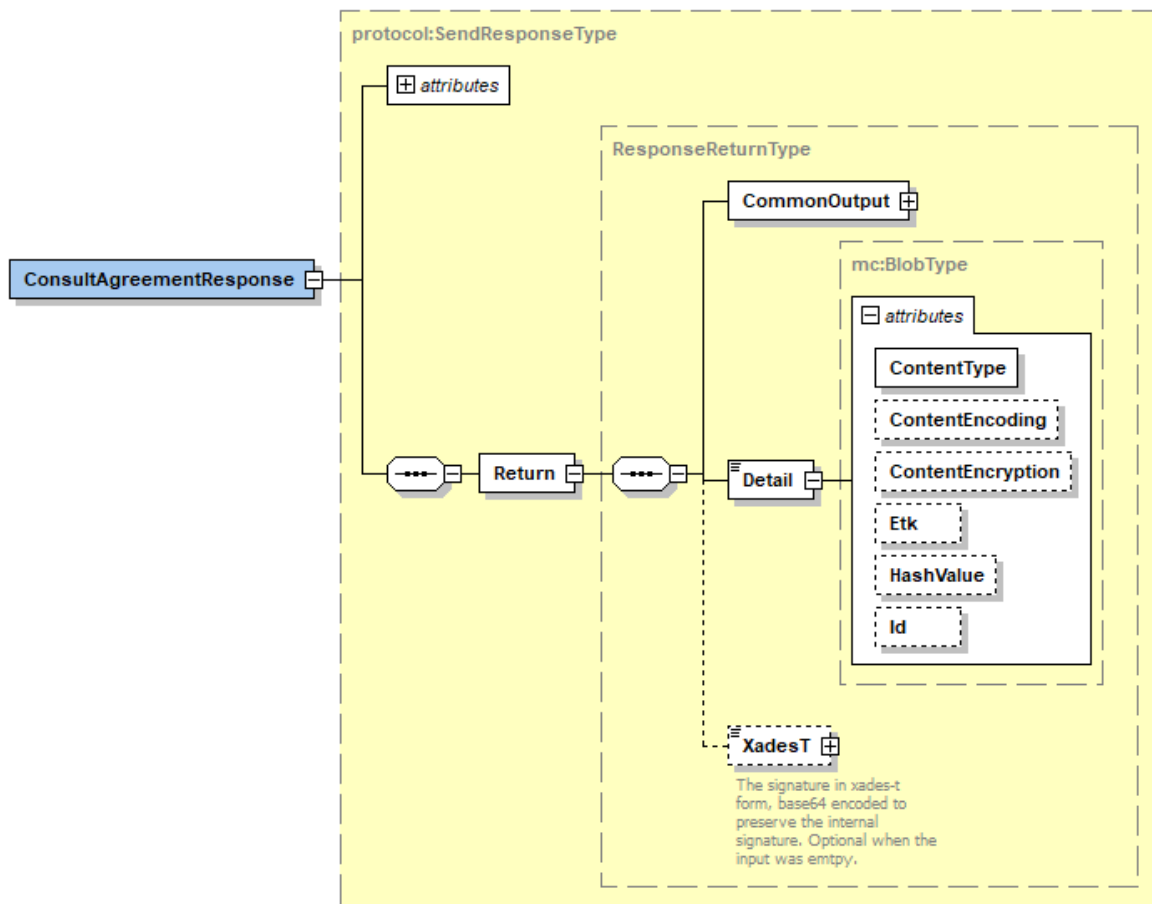
Business example is generated from the documentation

- 'Draft FHIR messages 20200731'
- 'Implementation Guide 20200731'

provided by CIN/NIC.



5.3.2.3 Output arguments in ConsultAgreementResponse



Field name	Description
"Response"	<p>@Id: Unique Id for tracing</p> <p>@InresponseTo: 'Id' attribute of the request if available</p> <p>@IssueInstant: Generation response moment</p>
Return	<p>See the documentation provided by the CIN/NIC for more details :</p> <ul style="list-style-type: none"> - 'Service_Catalogue_iSocial_GenSync' - 'Draft FHIR messages 20200731' - 'Implementation Guide 20200731' <p>NB: In this case, the attribute <code>@ContentEncryption</code> can only have the value "encryptedForKnownRecipient" (the content of the body is encrypted with the public key of the health-care provider).</p>

5.3.2.4 Response example

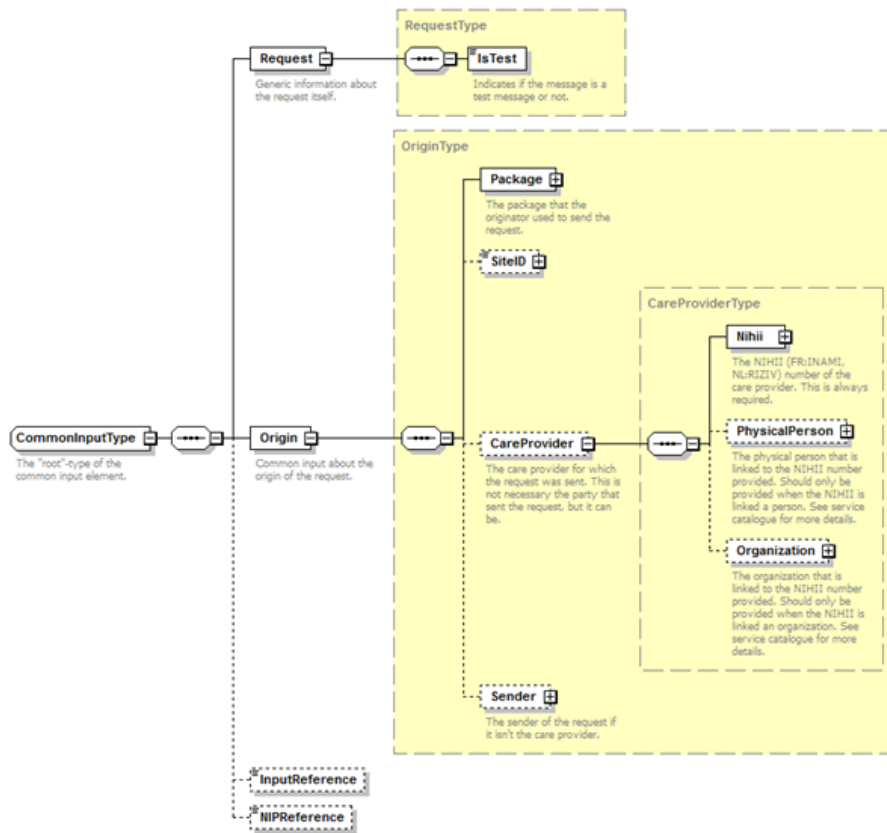
Business example is generated from the documentation

- 'Draft FHIR messages 20200731'
- 'Implementation Guide 20200731'

provided by CIN/NIC.

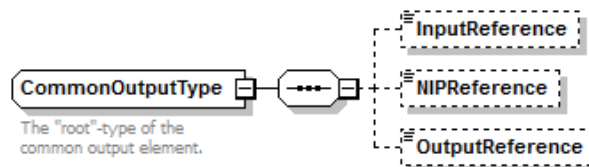
5.3.3 Used Types

5.3.3.1 CommonInputType



For the semantics of the particular elements and other information about the service, see the documentation [Service_Catalogue_iSocial_Commons](#) and [MyCareNet Authentication Catalogue](#) provided by the CIN/NIC.

5.3.3.2 CommonOutputType



For the semantics of the particular elements and other information about the service see the documentation [Service_Catalogue_iSoacial_Commons](#) provided by the CIN/NIC

6 Security

6.1 Business security

In case the development adds an additional use case based on an existing integration, the eHealth platform must be informed at least one month in advance with a detailed estimate of the expected load. This will ensure an effective capacity management.

In case of technical issues on the WS, the partner may obtain support from the contact center (see Chap 3).

In case the eHealth platform finds a bug or vulnerability in its software, we advise the partner to update his application with the newest version of the software within 10 business days.

In case the partner finds a bug or vulnerability in the software or web service that the eHealth platform delivered, he is obliged to contact and inform us immediately. He is not allowed to publish this bug or vulnerability in any case.

6.2 Web service

WS security used in this manner is in accordance with the common standards. Your call will provide:

- SSL one way
- Time-to-live of the message: one minute. Note that the time-to-live is the time difference between the Created and Expires elements in the Timestamp and is not related to the timeout setting on the eHealth ESB, etc. This means that eHealth will process the message if it is received within the time-to-live value (there is also tolerance of 5 minutes to account for the clock skew), but the actual response time may be greater than one minute in some situations.
- Signature of the timestamp and body. This will allow eHealth to verify the integrity of the message and the identity of the message author.

7 Test and release procedure

7.1 Procedure

This chapter explains the procedures for testing and releasing an application in acceptance or production.

7.1.1 Initiation

If you intend to use the eHealth service in the acceptance environment, please contact info@ehealth.fgov.be. The Project department will provide you with the necessary information and mandatory documents.

7.1.2 Development and test procedure

You have to develop a client in order to connect to our web service. Most of the required integration info to integrate is published in the technical library on the eHealth portal.

In some cases, the eHealth platform provides you with a mock-up service or test cases in order for you to test your client before releasing it in the acceptance environment.

7.1.3 Release procedure

When development tests are successful, you can request to access the eHealth acceptance environment.

From this moment, you can start integration and acceptance tests. The eHealth platform suggests testing during minimum one month.

After successful acceptance tests, the partner sends his test results and performance results with a sample of “eHealth request” and “eHealth answer” to the eHealth point of contact by email.

Then the eHealth platform and the partner agree on a release date. The eHealth platform prepares the connection to the production environment and provides the partner with the necessary information. During the release day, the partner provides the eHealth platform with feedback on the test and performance tests.

For further information and instructions, please contact: integration-support@ehealth.fgov.be.

7.1.4 Operational follow-up

Once in production, the partner using the eHealth service for one of his applications will always test first in the acceptance environment before releasing any adaptations of his application in production. In addition, he will inform eHealth on the progress and test period.

7.2 Test cases

eHealth recommends performing tests for the following case:

- AskAgreement (contact NIC/CIN for test data of the patients)
- ConsultAgreement

In addition, the organisation should also run negative test cases.



8 Error and failure messages

There are different possible types of response:

- If there are no technical errors, responses as described in section 5 are returned.
- In the case of a technical error, a SOAP fault exception is returned (see table below).

If an error occurs, first please verify your request. Following table contains a list of common system error codes for the eHealth Service Bus. For possible business errors, refer to the documentation 'GenericSync Error codes' and 'Service_Catalogue_Commons' provided by the CIN/NIC.

Table 1: Description of the possible SOAP fault exceptions.

Error code	Component	Description	Solution/Explanation
SOA-00001		Service error	This is the default error sent to the consumer in case more details are unknown.
SOA-01001	Consumer	Service call not authenticated	From the security information provided; <ul style="list-style-type: none"> • or the consumer could not be identified • or the credentials provided are not correct
SOA-01002	Consumer	Service call not authorized	The consumer is identified and authenticated, but is not allowed to call the given service.
SOA-02001	Provider	Service not available Please contact service desk	<ul style="list-style-type: none"> • An unexpected error has occurred; • Retries will not work; • Service desk may help with root cause analysis.
SOA-02002	Provider	Service temporarily not available Please try later	<ul style="list-style-type: none"> • An unexpected error has occurred; • Retries should work; • If the problem persists service desk may help.
SOA-03001	Consumer	Malformed message	This is the default error for content related errors in case more details are unknown.
SOA-03002	Consumer	Message must be SOAP	Message does not respect the SOAP standard.
SOA-03003	Consumer	Message must contain SOAP body	Message respects the SOAP standard, but body is missing.
SOA-03004	Consumer	WS-I compliance failure	Message does not respect the WS-I standard.
SOA-03005	Consumer	WSDL compliance failure	Message is not compliant with WSDL in Registry/Repository.
SOA-03006	Consumer	XSD compliance failure	Message is not compliant with XSD in Registry/Repository.
SOA-03007	Consumer	Message content validation failure	From the message content (conform XSD): <ul style="list-style-type: none"> • Extended checks on the element format failed; • Cross-checks between fields failed.

If the cause is a business error, please contact MyCareNet at ServiceDesk@MyCareNet.be.

The soap header (only when the received response is not a SOAP fault) contains a message ID, e.g.:

```
<soapenv:Header>
```

```
  <add:MessageID
```

```
xmlns:add="http://www.w3.org/2005/08/addressing">6f23cd40-09d2-4d86-b674-  
b311f6bdf4a3</add:MessageID>
```

```
</soapenv:Header>
```

This message ID is important for tracking of the errors so when available, please provide it when requesting support.

