

**EMSR WS v1 Registration
Cookbook
Version 2.14**

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. Support 7**
- 3.1 Helpdesk eHealth platform 7
- 3.1.1 Certificates..... 7
- 3.1.2 For issues in production 7
- 3.1.3 For issues in acceptance 7
- 3.1.4 For business issues 7
- 3.2 Status 7
- 3.3 End-to-End Encryption 7
- 4. Global overview 9**
- 5. Step-by-step 10**
- 5.1 Technical requirements 10
- 5.2 Use of the eHealth SSO solution 10
- 5.2.1 Ambulance service..... 10
- 5.2.2 Security policies to apply 10
- 5.3 Web service..... 11
- 5.3.1 Method RegisterPartA & RegisterPartB 12
- 5.3.2 Method *GetSds*..... 17
- 5.3.3 Used Types..... 24
- 5.3.4 DeleteSheet 26
- 5.3.5 AnonymizePatientData 28
- 6. Risks and security..... 32**
- 6.1 Security 32
- 6.1.1 Business security 32
- 6.1.2 Web service 32
- 6.1.3 The use of username, password and token..... 32
- 7. Test and release procedure 33**
- 7.1 Procedure..... 33
- 7.1.1 Initiation 33
- 7.1.2 Development and test procedure 33
- 7.1.3 Release procedure 33



7.1.4	Operational follow-up	33
7.2	Test cases	33
8.	Error and failure messages.....	35
8.1	Business errors.....	35
8.1.1	Status Code.....	35
8.1.2	Status Detail.....	36
8.2	Technical errors.....	38

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	28/07/2016	SMALS	First version
1.1	10/09/2016	eHealth platform	Fine tuning Consultation & Registration Added: chapter 5.1.1. (SSO)
1.2	18/10/2016	SMALS	Change certificate identifier
2.0	31/08/2017	SMALS	Change UAM Configuration and KMEHR message structure
2.1	20/09/2017	SMALS	Corrections after feedback from integrator
2.2	24/10/2017	SMALS	Correction of getSDS to getSds.
2.3	12/12/2017	SMALS	Small correction to comply with version 2.0.12 of EMSR
2.4	19/01/2018	SMALS	Adaptation of error messages (version 2.0.13 of EMSR).
2.5	22/05/2018	SMALS	Minor changes in the KMEHR. Corresponds with v2.0.15 of EMSR.
2.6	21/06/2018	SMALS	Main change in this version is the public key (etk) that has to be used to encrypt requests to EMSR. This corresponds with version v2.0.16 of EMSR.
2.7	02/10/2018	SMALS	Minor changes related to release 2.0.20.
2.8	12/10/2018	SMALS	Changes in the KMEHR Cookbook related to release 2.0.22 and TLS configuration.
2.9	13/12/2018	SMALS	Modification of RegisterPartB in accordance to Release 2.0.24
2.10	07/02/2019	SMALS	Modification of KMEHR related to field IDs.
2.10.1	11/07/2019	SMALS	Modification after feedback from eHealth
2.11	22/10/2019	SMALS	Release 2.0.35 of EMSR – addition of x and y coordinates in the GetSDS method
2.12	31/07/2020	SMALS	New method DeleteSheet and AnonymizePatientData
2.13	31/12/2020	SMALS	Correction/New fields for method GetSDS + editable destination hospital part B
2.14	22/01/2024	SMALS	Register part A & B shorter registration deadline

2. Introduction

2.1 Goal of the service

The purpose of this service is to provide authenticated ambulances services and hospitals with a set of methods for registering and consulting EMSR (Emergency Medical Service Registry) sheets and consulting SDS data.

The GetSheetList and GetSheet method will be available for hospitals and ambulances services.

The RegisterPartA, RegisterPartB, GetSds and DeleteSheet method will be available for ambulances services.

The sheet registering is a 2-step process.

1. PartA is sent when a patient arrives at an hospital. It contains any information about the patient available at the time of arrival.
2. PartB is sent within 5 days. This step will provide the complete identification information about the patient.

On the sheet consultation the system concatenates partA, partB and SDS data (most recent SDS data) to generate the intervention sheet. Patient information from partB will replace the patient information of part B.

It is possible to separately get SDS data with the GetSds method in order to retrieve address for example.

2.2 Goal of the document

In this cookbook, we explain the structure and content aspects of the possible requests and the replies of EMSR WS. An example illustrates each of those messages. In addition, a list of possible errors can be found in this document.

This information should allow (the IT department of) an organization to develop and use the WS call.

Some technical and legal requirements must be met in order to allow the integration of EMSR WS in client applications.

This document is neither a development nor a programming guide for internal applications; eHealth partners always keep a total freedom within those fields. Nevertheless, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth partners must commit to comply with specifications, data format, and release processes described within this document. In addition, our partners in the health sector must also comply with the business rules of validation and integration of data within their own applications in order to minimize errors and incidents.

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	EMSR v1 - KMEHR Message Cookbook	3.0	25/03/2021	HFCSE
2	Glossary.pdf	1.0	01/01/2010	eHealth platform
3	STS Cookbook	1.6	25/01/2023	eHealth platform

¹ www.ehealth.fgov.be

4	Cookbook end-to-end vercijfering voor bekende bestemming / encryption end- to-end vers destinataire connu	2.9	18/07/2022	eHealth platform
---	--	-----	------------	------------------

2.4 External document references

All documents can be found through the internet. They are available to the public, but not supported by the eHealth platform.

ID	Title	Source	Date	Author
1	OASIS – Web services security – SAML Token Profile 1.1	https://www.oasis-open.org/committees/download.php/16768/wssv1.1-spec-os-SAMLSAMLTokenProfile.pdf	01/02/2006	OASIS Standard
2	Lijst met de AMBUREG- variabelen Liste des variables AMBUREG	https://www.health.belgium.be/nl/richtlijnen-ambureg https://www.health.belgium.be/fr/directives-ambureg	20/05/2019	FOD Volksgezondheid, Veiligheid van de Voedselketen en Leefmilieu SPF Santé Publique, Sécurité de la chaîne alimentaire et Environnement

3. Support

3.1 Helpdesk eHealth platform

3.1.1 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

3.1.2 For issues in production

eHealth platform contact centre:

- Phone: 02 788 51 55 (on working days from 7 am till 8 pm)
- 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)

3.1.3 For issues in acceptance

Integration-support@ehealth.fgov.be

3.1.4 For business issues

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

3.2 Status

The website <https://status.ehealth.fgov.be> is the monitoring and information tool for the ICT functioning of the eHealth services that are partners of the Belgian eHealth system.

3.3 End-to-End Encryption

In order to secure the information exchanged between ambulances/hospitals and EMSR WS, most of requests and responses contain encrypted data. Encryption is performed with a public encryption key belonging to the recipient. This means that a request sent to EMSR WS has to use EMSR public encryption token and the response will use the public encryption key of the ambulance/hospital that sent the request to encrypt the data (The public encryption token key is required in the request). eHealth platform provides the procedures to create a pair of private/public keys and the public encryption token.

<urn:Value>1990003302</urn:Value>

- Process of encryption upon sending of data (RegisterPartA, RegisterPartB):
 - 1) Sign the KMEHR data to encrypt with your personal private key



- 2) Get the public key of The Federal Public Service of Health (EHP : 1990003302) with application id 'EMSR' from the ETK Depot.

The GetEtk Request should contain following search criteria:

```
<urn:Identifier>
  <urn:Type>EHP</urn:Type>
  <urn:Value>1990003302</urn:Value>
  <urn:ApplicationID>EMSR</urn:ApplicationID>
</urn:Identifier>
```

- 3) Encrypt the data with the public key above and using eHealth encryption libraries.
 - 4) Sign again with your personal private key
- Process of decryption upon consulting of data (GetSheet):
 - 1) Verify validity of signature
 - 2) Decrypt KMEHR data using your private key.

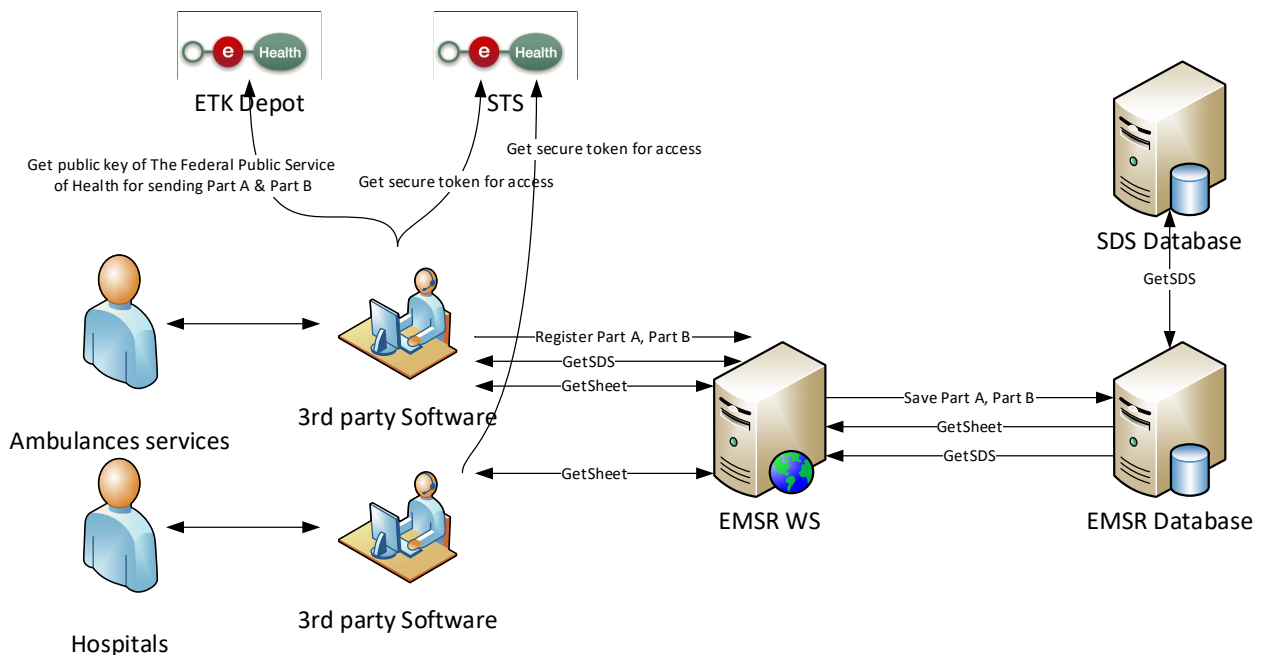
Therefore, the GetSheet request has to contain your public key which will be used to encrypt the GetSheet response.

In XML, the encrypted data is represented in base64 binary format, translating each byte of binary data into an ASCII string format. Thus from XML data, this representation has first to be decoded into bytes using the base64 encoding scheme before being decrypted.

Note: While the base64 files are in ASCII format, they should still be encoded using the UTF-8 format.



4. Global overview



This global overview aims to show how the registration WS is used.

- Step 1. To use the Registration WS, you have to contact the WS STS to get a secure token containing the identification of the user (see 5.2 Use of the eHealth SSO solution and the STS Cookbook).
- Step 2. To encrypt the content of your message, you have to call method Get ETK of the ETK Depot Web Service (synonymous to the ETEE Webservice, **see ETEE for known recipient Cookbook on the eHealth Portal**) in order to get the public key of the recipient and use the Crypto Library.
See 3.3 End-to-End Encryption for more detailed information.
- Step 3. Once you have your secure token, you are able to use and contact the Registration service to register part A or part B.
- Step 4. When your call has been sent, the system will respond with a response message.

5. Step-by-step

5.1 Technical requirements

All the xml requests submitted to the WS must be encoded in the UTF-8 format.

5.2 Use of the eHealth SSO solution

For each WS accessed on eHealth platform, authentication ensures that the requester is allowed to do so. The eHealth certificates are used to trust the requester. In order to use EMSR Registration, prior authentication has to be made on STS with the use of the eHealth Certificate and with specific parameters. An assertion will be generated and used to call and access the EMSR Registration service.

The complete overview of the profile and a systematic implementation to start protecting a new application with SSO @ eHealth is described in the eHealth STS cookbook.

In order to implement a call to the eHealth STS you can reuse the implementation as provided in the "eHealth technical connector":

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

Nevertheless, eHealth implementations use standards and any other compatible technology (WS stack for the client implementation) can be used instead.

The attributes that need to be provided and the attributes that should be certified by eHealth in order to obtain a token valid for EMSR Registration services are described in following section. To access the EMSR Registration WS, the response token must contain "true" for all of the certification attributes. If you obtain "false", contact eHealth to verify whether the requested test cases were correctly configured (See section 3).

5.2.1 Ambulance service

The SAML token request is secured with the eHealth certificate of the ambulance service. The certificate used by the Holder-Of-Key (HOK) verification mechanism is the same eHealth certificate. The required attributes are the following (AttributeNamespace="urn:be:fgov:identification-namespace"):

- The NIHII number of the ambulance service:
urn:be:fgov:health:1.0:certificateholder:ambulanceservice:nihii-number and
urn:be:fgov:health:1.0:ambulanceservice:nihii-number

You must also specify which information must be asserted by the eHealth platform:

- The NIHII number of the ambulance service (AttributeNamespace="urn:be:fgov:identification-namespace"): *urn:be:fgov:health:1.0:certificateholder:ambulanceservice:nihii-number* and
urn:be:fgov:health:1.0:ambulanceservice:nihii-number
- The ambulance service must be a recognized ambulance service (AttributeNameSpace="urn:be:fgov:certifiednamespace:ehealth"): *urn:be:fgov:health:1.0:certificateholder:ambulanceservice:nihii-number:recognisedambulanceservice:boolean*

5.2.2 Security policies to apply

See section 3.2 for the update in the TLS configuration.

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



As WS security policy, we expect:

- A timestamp (the date of the request), with a time to live of one minute. (if the message does not arrive during this minute, it shall not be treated).
- The signature with the certificate of
 - the timestamp, (the one mentioned above)
 - the body (the message itself)
 - and the binary security token: an eHealth certificate or a SAML token issued by STS

This will allow the eHealth platform to verify the integrity of the message and the identity of the message author.

A document explaining how to implement this security policy can be obtained at the eHealth platform. This document, the STS cookbook, can be found on the eHealth portal.

5.2.3 WS-I Basic Profile 1.1

Your request must be WS-I compliant (See Chap 2.4 - External Document Ref).

[If not applicable, you can delete this Par.](#)

5.2.4 Tracing

To use this service, the request SHOULD contain the following two http header values (see RFC

<https://datatracker.ietf.org/doc/html/rfc7231#section-5.5.3>):

1. User-Agent: information identifying the software product and underlying technical stack/platform. It MUST include the minimal identification information of the software such that the emergency contact (see below) can uniquely identify the component.
 - a. Pattern: {minimal software information}/{version} {minimal connector information}/{connector-package-version}
 - b. Regular expression for each subset (separated by a space) of the pattern: `[[a-zA-Z0-9-\\]]*\\[[0-9azA-Z-_.]]*`
 - c. Examples:
User-Agent: myProduct/62.310.4 Technical/3.19.0
User-Agent: Topaz-XXXX/123.23.X freeconnector/XXXXX.XXX
2. From: email-address that can be used for emergency contact in case of an operational problem.
Examples:
From: info@mycompany.be

5.3 Web service

This WS handles the collection of data for an intervention. There are four different methods on the Registration WS. Two methods will be used to register the part A and the part B of the intervention, one will allow users to get one or more SDS records and finally one will let the user to delete an existing intervention.

The content of KMEHR part is described in the file “ambureg_variables_vxx.xlsx”.

The EMSR Registration webservice has the following endpoints:

- Integration environment: <https://services-int.ehealth.fgov.be/EMSR/Registration/v1>
- Acceptance environment: <https://services-acpt.ehealth.fgov.be/EMSR/Registration/v1>
- Production environment: <https://services.ehealth.fgov.be/EMSR/Registration/v1>



5.3.1 Method RegisterPartA & RegisterPartB

This method is a collecting method for the first and the second part of an intervention. It will take a KMEHR as input and will acknowledge the reception of the message. If the KMEHR is not correct, an error will be returned with the invalidated rule and an explanation of the rule.

When 'Registering' the different parts, always integrate the variables of the **patient** (under folder element) already sent in part A in the KMEHR message header. The MissionIdentification number is not mandatory when registering the part A of the sheet. In the part B however, it is **mandatory** and the mission must be assigned to the team that registers the part B.

Once the part B has been sent, the sheet is considered complete. Only the patient and the hospital of destination can be changed after PartB is sent. Beware that changing the hospital of destination can only be done once.

The registration of part A or part B can only be done up to 3 months after the call date (call date can be found in the SDS).

It is accomplished by correcting the following fields in a registerPartB request:

- 29.1 patient.patientNationalNumber
- 29.2 patientName
- 30 patientFirstname
- 31.1 patient.patientInterventionAddrFlag
- 31.4 patient.patientAddressAbroadFlag
- 31.4.1 patient.patientAddressCountryCode
- 31.5 patient.patientZipCode
- 31.6 patient.patientLocality
- 31.8 patient.patientStreet
- 31.9 patient.patientStreetNo
- 31.10 patient.patientbox
- 31.10.1 patientPatientForeignAdress/FreeText
- 31.11 invoicingaddresspatientaddressFlag
- 31.12 invoicing.name
- 31.13 invoicing.Firstname
- 31.14 invoicing.Street
- 31.15 invoicing.Streetno
- 31.16 invoicing.Zipcode
- 31.17 invoicing.Locality
- 31.18 invoicing.box
- 31.19 invoicing.AbroadFlag
- 31.19.1 invoicing.CountryCode
- 31.20 invoicing.abroadFreeText
- 32.1 patient.patientDob
- 33 patient.patientSexCode

For the hospital of destination, the following fields are editable once:

- 41.1 hospitalcode



- 41.1.1 hospitalsiteas

The folder also needs to contain the correct namespaces:

- xmlns="http://www.ehealth.fgov.be/standards/kmehr/schema/v1"
- xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
- xsi:schemaLocation="http://www.ehealth.fgov.be/standards/kmehr/schema/v1 xsd-kmehr-1.17.0/ehealth-kmehr/XSD/kmehr_elements-1_17.xsd">

```
<folder xmlns="http://www.ehealth.fgov.be/standards/kmehr/schema/v1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ehealth.fgov.be/standards/kmehr/schema/v1
xsd-kmehr-1.17.0/ehealth-kmehr/XSD/kmehr_elements-1_17.xsd">
  <id S="ID-KMEHR" SV="1.0">1</id>
  <patient>
    <id S="INSS" SV="1.0">65406506540</id>
    ...
  </patient>
</folder>
```

5.3.1.1 Request

The request mainly consists of one element: the encrypted KMEHR message.

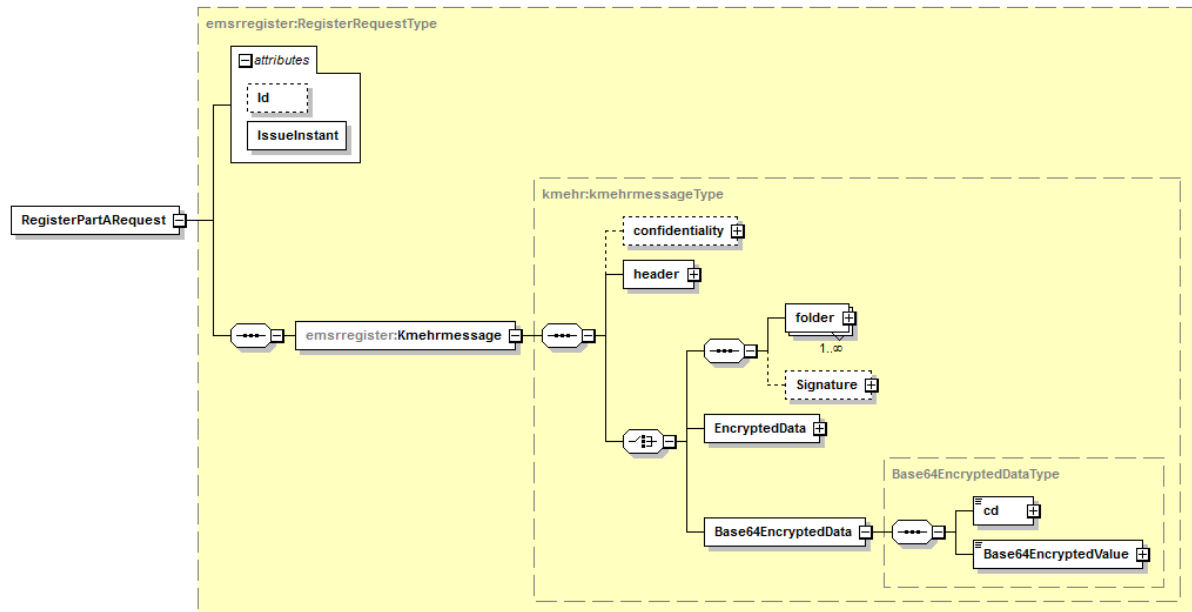


Figure 1: RegisterPartARequest

Field Name	Description	Attribute	Required
Id	Identifier of the request within the caller system.	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
Kmehrmessage	The KMEHR message contains at least header and folder. The folder is contained in the Base64EncryptedData, while the header is not crypted in the KMEHR message.	No	Yes
Base64EncryptedValue	This element contains a KMEHR message, encrypted using ETEE (see ETEE CookBook) and encoded Base64. The Base64EncryptedValue contains <u>only</u> the <u>folder</u> element of the KMEHR message.	No	Yes

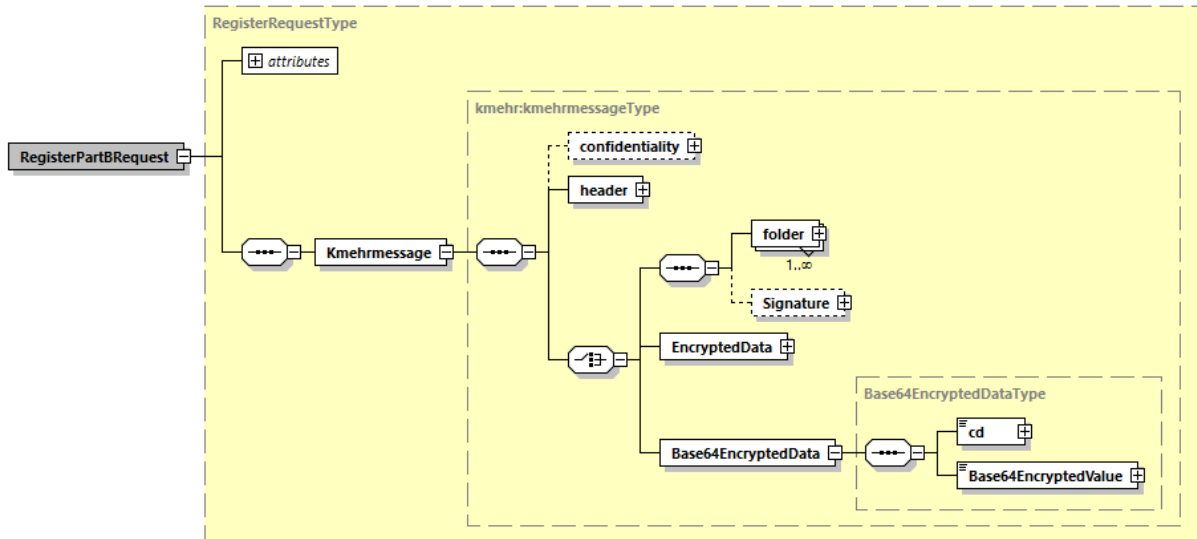


Figure 2: RegisterPartBRequest

Id	Identifier of the request within the caller system.	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
Kmehrmessage	The KMEHR message contains at least header and folder. The folder is contained in the Base64EncryptedData , while the header is not encrypted in the Kmehrmessage.	No	Yes
Base64EncryptedValue	This element contains a KMEHR message, encrypted using ETEE (see ETEE CookBook) and encoded Base64. The Base64EncryptedValue contains <u>only</u> the <u>folder</u> element of the KMEHR message.	No	Yes

5.3.1.2 Response

The response is an acknowledgement. The status element description is detailed in 5.4.4.1 StatusType.

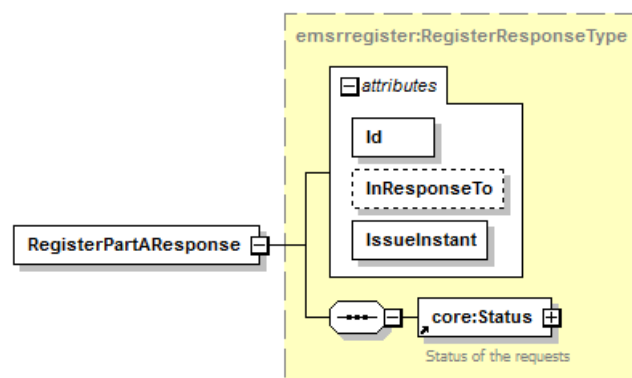


Figure 3: RegisterPartAResponse

Field Name	Description	Attribute	Required
Id	Identifier of the response.	Yes	Yes

InResponseTo	Id attribute of the request	Yes	No
IssueInstant	Date and time of the response.	Yes	Yes
Status	See section 5.4.3.1 StatusType	No	Yes

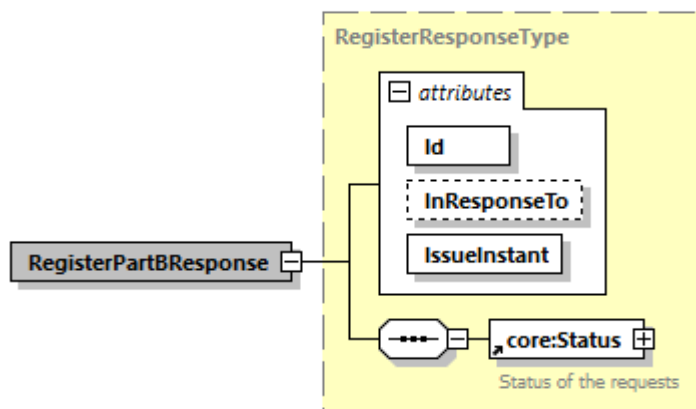


Figure 4: RegisterPartAResponse

Field Name	Description	Attribute	Required
Id	Identifier of the response.	Yes	Yes
InResponseTo	Id attribute of the request	Yes	No
IssueInstant	Date and time of the response.	Yes	Yes
Status	See section 5.4.3.1 StatusType	No	Yes

5.3.1.3 Example

Request:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soap:Body>
    <emsregister:RegisterPartARequest Id="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3" IssueInstant="2001-12-17T09:30:47Z"...>
      <ecore:Kmehrmessage>
        <!-- Unencrypted part of the KMEHR (contains the header of the KMEHR) -->
        <kmehr:header> see KMEHR CookBook </kmehr:header>
        <!-- Encrypted part of the KMEHR (contains the folder of the KMEHR)-->
        <kmehr:Base64EncryptedData>
          <kmehr:cd S="CD-ENCRYPTION-METHOD" SV="1.0">CMS</cd>
        </kmehr:Base64EncryptedData>
        <kmehr:Base64EncryptedValue>fAkEnUmBeRrrrrrjBsR09EbGhjZ0dTQUxNQUFBUUNBRU1tQ1p0dU1GUXhEUzhi</kmehr:Base64EncryptedValue>
        </kmehr:Base64EncryptedData>
      </ecore:Kmehrmessage>
    </emsregister:RegisterPartARequest>
  </soap:Body>
</soap:Envelope>
```



```
</soap:Body>
</soap:Envelope>
```

Response:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <emsrregister:RegisterPartAResponse Id="_de2754ca-83fe-41ce-
9c72-9c3a7f586b38" InResponseTo="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3"
IssueInstant="2016-04-07T10:09:48.288+02:00" ...>
      <core:Status>
        <core:StatusCode
Value="urn:be:fgov:ehealth:2.0:status:Success"/>
      </core:Status>
    </emsrregister:RegisterPartAResponse>
  </soap:Body>
</soap:Envelope>
```

5.3.2 Method *GetSds*

The *GetSds* method retrieves all the SDS messages related to a specific mission identification. The User's entity will be used implicitly during the search.

These SDS provide in particular information concerning the timings of the interventions (recorded via a radio), as well as certain data of the intervention (which has passed the call to the 100 number, where the intervention took place...).

5.3.2.1 Request

Pagination

In the request *Offset* and *MaxElements* attributes must be specified. *Offset* is set to 0, and *MaxElements* can be maximum 100. When 100 elements are returned, this can mean more results are present. A second call can be made with *Offset* set to 100 and *MaxElements* to 100 to gather other results. This can also mean that search criteria were not correctly selected.



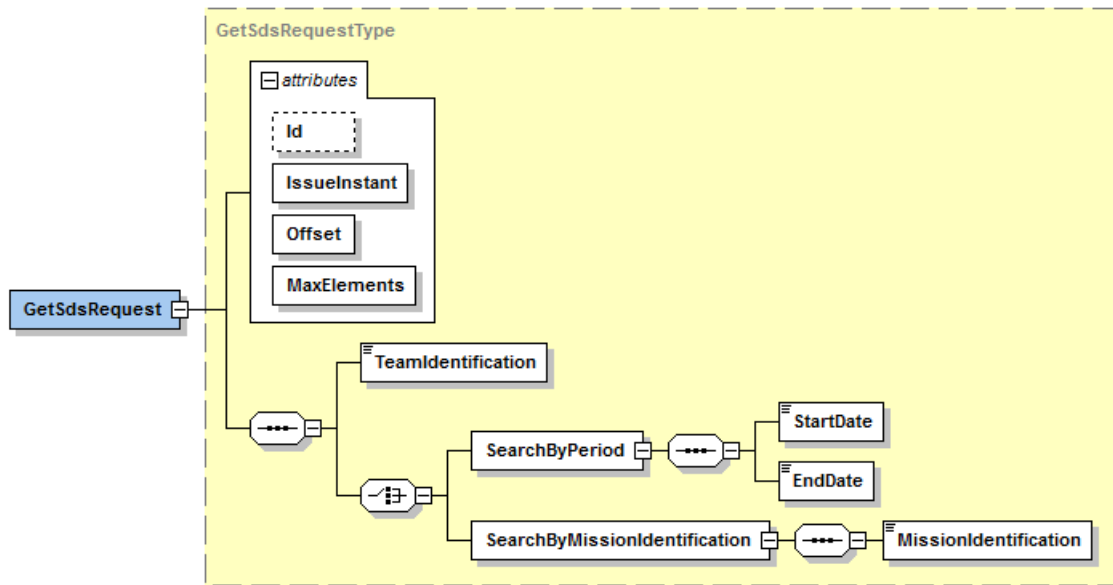


Figure 5: GetSdsRequest

Field Name	Description	Attribute	Required
Id	Identifier of the request within the caller system.	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
Offset	Index of first element of response	Yes	Yes
MaxElements	Number of result per request (max 100)	Yes	Yes
StartDate	Beginning of the date range search	No	Yes
EndDate	End of the date range search	No	Yes
TeamIdentification	This is the unique identification number of the team in 9 characters.	No	Yes
MissionIdentification	Id of the mission	No	Yes

5.3.2.2 Response

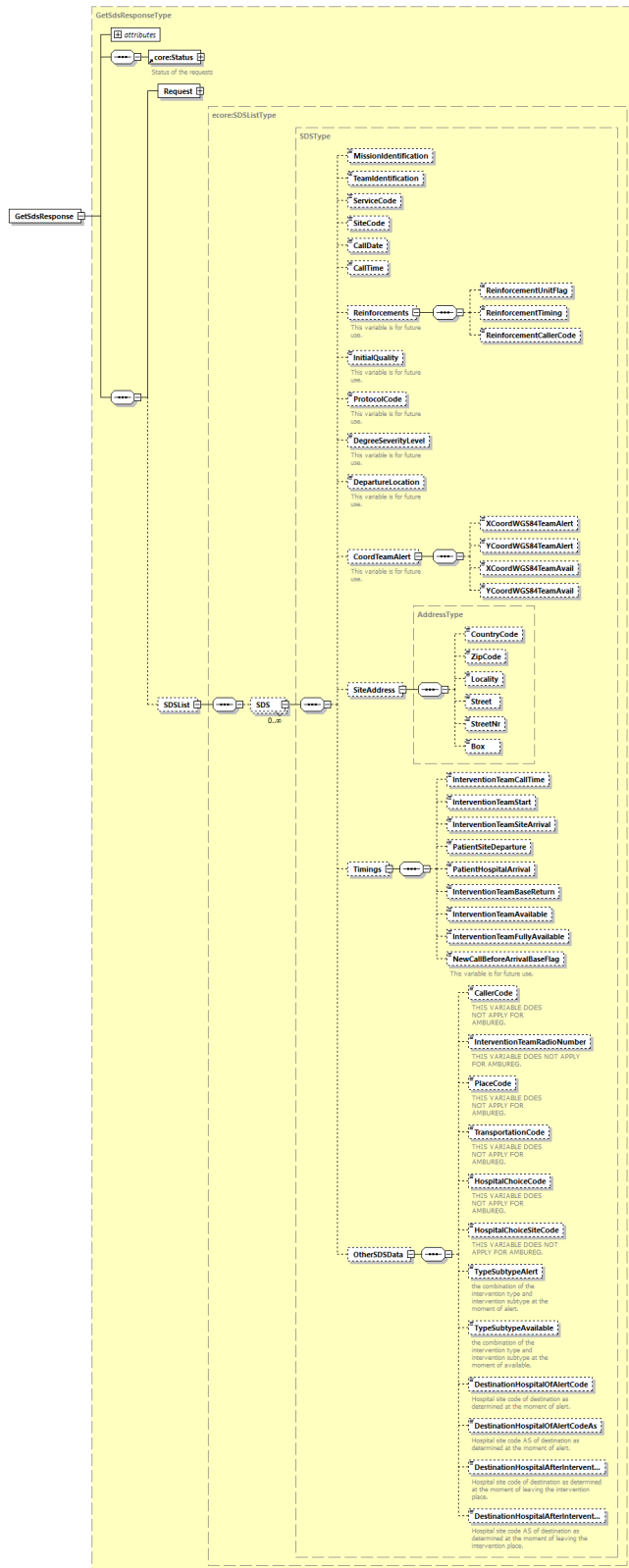


Figure 6: GetSdsResponse



Field Name	Description	Attribute	Required
Id	Identifier of the request within the caller system.	Yes	No
InResponseTo	Id attribute of the request	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
Offset	Index of first element of response	Yes	Yes
MaxElements	Number of result per request (max 100)	Yes	Yes
Status	See section 5.4.3.1 StatusType	No	Yes
MissionIdentification	Mission dispatch number	No	No
TeamIdentification	Identification of the team which carried out the intervention	No	No
ServiceCode	The approval number of the service	No	No
SiteCode	Departure site of the day	No	No
CallDate	Date of the call	No	No
CallTime	Time of the call	No	No
CallerCode (not for this project)	Indicates who made the 100 call. See values below. THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
ReinforcementUnitFlag	This variable is for future use.	No	No
ReinforcementTiming	This variable is for future use.	No	No
ReinforcementCallerCode	This variable is for future use.	No	No
InitialQuality	This variable is for future use.	No	No
ProtocolCode	This variable is for future use.	No	No
DegreeSeverityLevel	This variable is for future use.	No	No
DepartureLocation	This variable is for future use.	No	No
XCoordWGS84TeamAlert	This variable will contain the X coordinate of the intervention address.	No	No
YCoordWGS84TeamAlert	This variable will contain the Y coordinate of the intervention address.	No	No
XCoordWGS84TeamAvail	This variable is for future use.	No	No
YCoordWGS84TeamAvail	This variable is for future use.	No	No
SiteAddress	Intervention address	No	No
InterventionTeamCallTime	Date and time of the first ring tone of the call.	No	No
InterventionTeamStart	Departure date and time of the intervention team.	No	No



InterventionTeamSiteArrival	Arrival date and time on site.	No	No
PatientSiteDeparture	Departure date and time of the intervention team from the site.	No	No
PatientHospitalArrival	Arrival date and time of the intervention team at the hospital.	No	No
InterventionTeamBaseReturn	Date and time of the intervention team returns to base.	No	No
InterventionTeamAvailable	Date and time of when the unit goes back to available status.	No	No
InterventionTeamFullyAvailable	Date and time of when the unit is really available at quarter again, after cleaning, replacement of material, etc.	No	No
NewCallBeforeArrivalBaseFlag	This variable is for future use.	No	No
InterventionTeamRadioNumber	THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
PlaceCode (not for this project)	Identifies the type of intervention place. THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
TransportationCode (not for this project)	Code identifying the type of transport. THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
HospitalChoiceCode	THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
HospitalChoiceSiteCode (not for this project)	This indicates why the patient was transported to the mentioned hospital. THIS VARIABLE DOES NOT APPLY FOR AMBUREG.	No	No
TypeSubtypeAlert	The combination of the intervention type and intervention subtype at the moment of alert.	No	No
TypeSubtypeAvailable	The combination of the intervention type and intervention subtype when again available.	No	No
DestinationHospitalOfAlertCode	Hospital site code of destination as determined at the moment of alert.	No	No
DestinationHospitalOfAlertCodeAs	Hospital site code AS of destination as determined at the moment of alert.	No	No
DestinationHospitalAfterInterventionCode	Hospital site code of destination as determined at the moment of leaving the intervention place.	No	No

DestinationHospitalAfterInterventionCodeAs	Hospital site code AS of destination as determined at the moment of leaving the intervention place	No	No
--	--	----	----

CallerCode:

- 1: doctor
- 2: SMUR
- 3: ambulance
- 4: hospital
- 5: home
- 6: police
- 7: fire brigade
- 8: private individual
- 9: other
- 10: unknown
- 11: PIT

5.3.2.3 Example

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:be:fgov:ehhealth:emsr:registration:v1">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:GetSdsRequest Id="_ded542e4-942b-4062-999b-14f20068761f" IssueInstant="2019-10-07T10:09:48.288+02:00" MaxElements="10" Offset="1">
      <urn:TeamIdentification>MOAZNI301</urn:TeamIdentification>
      <urn:SearchByPeriod>
        <urn:StartDate>2019-09-12</urn:StartDate>
        <urn:EndDate>2019-10-13</urn:EndDate>
      </urn:SearchByPeriod>
    </urn:GetSdsRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns4:GetSdsResponse Id="_877ba285-404d-49e6-a24d-1e3bf4bc346f" InResponseTo="_ded542e4-942b-4062-999b-14f20068761f" IssueInstant="2019-10-22T10:36:33.044+02:00" MaxElements="10" Offset="1"
      xmlns:ns10="urn:be:fgov:ehhealth:monitoring:protocol:v2"
      xmlns:ns2="http://www.w3.org/2000/09/xmldsig#"
      xmlns:ns3="http://www.w3.org/2001/04/xmlenc#"
      xmlns:ns4="urn:be:fgov:ehhealth:emsr:registration:v1">
```



```

xmlns:ns5="urn:be:fgov:health:commons:core:v2"
xmlns:ns6="urn:be:fgov:health:commons:protocol:v2"
xmlns:ns7="urn:be:fgov:health:emsr:core:v1"
xmlns:ns8="http://www.health.fgov.be/standards/kmehr/schema/v1"
xmlns:ns9="urn:be:fgov:health:errors:service:v1">
<ns5:Status>
  <ns5:StatusCode Value="urn:be:fgov:health:2.0:status:Success"/>
</ns5:Status>
<ns4:SDSList>
  <ns7:SDS>
    <ns7:MissionIdentification>40192550015</ns7:MissionIdentification>
    <ns7:TeamIdentification>MOAZNI301</ns7:TeamIdentification>
    <ns7:CallDate>2019-09-12+02:00</ns7:CallDate>
    <ns7:CallTime>00:44:35.000+02:00</ns7:CallTime>
    <ns7:CoordTeamAlert>
      <ns7:XCoordWGS84TeamAlert>5.21072105741794</ns7:XCoordWGS84TeamAlert>
      <ns7:YCoordWGS84TeamAlert>31.1272523362156</ns7:YCoordWGS84TeamAlert>
    </ns7:CoordTeamAlert>
    <ns7:SiteAddress>
      <ns7:CountryCode>BE</ns7:CountryCode>
      <ns7:ZipCode>9140</ns7:ZipCode>
      <ns7:Locality>Temse</ns7:Locality>
      <ns7:Street>Streetname</ns7:Street>
      <ns7:StreetNr>1</ns7:StreetNr>
    </ns7:SiteAddress>
    <ns7:Timings>
      <ns7:InterventionTeamCall>2019-09-
12T00:47:44.000+02:00</ns7:InterventionTeamCall>
    </ns7:Timings>
    <ns7:OtherSDSData>
      <ns7:CallerCode>10</ns7:CallerCode>
      <ns7:PlaceCode>9</ns7:PlaceCode>
    </ns7:OtherSDSData>
  </ns7:SDS>
</ns4:SDSList>
</ns4:GetSdsResponse>
</soap:Body>
</soap:Envelope>

```

5.3.3 Used Types

5.3.3.1 StatusType

eHealth SOA service response is composed of an *Status* element. This element is used to indicate the status of the completion of the request. The status is represented by a *StatusCode* and optionally, the message describing the status. Additional detail gives extra information on the encountered business errors returned by the target service.

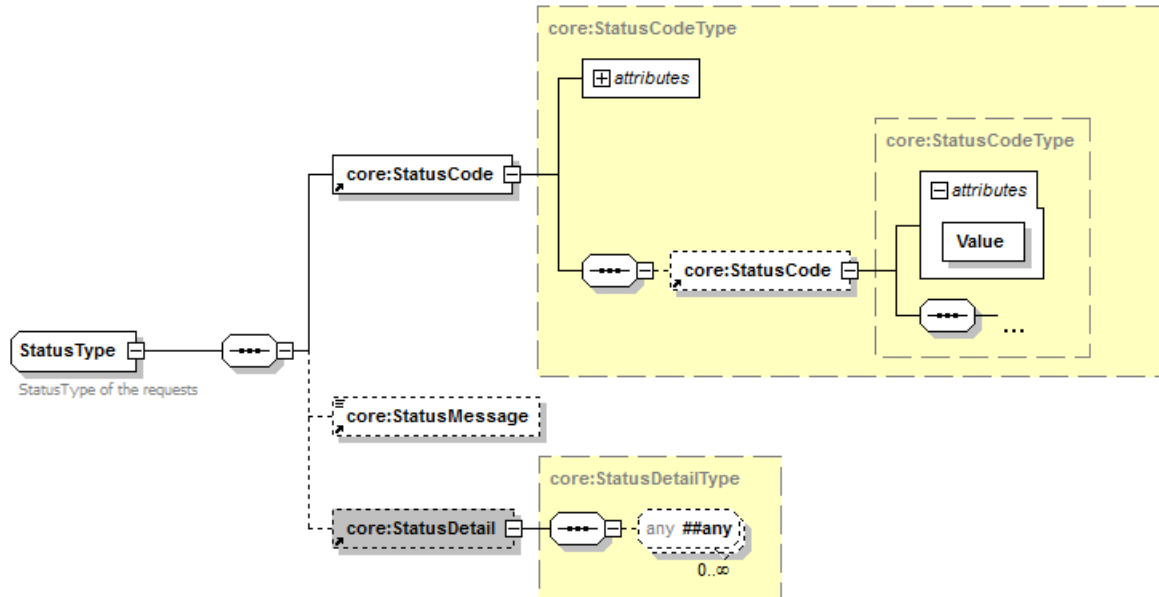


Figure 7: StatusType

Field Name	Description	Attribute	Required
StatusCode	See table further down for a list of possible values	No	Yes
StatusMessage	An optional message describing the error.	No	Yes
StatusDetail	The StatusDetail is defined as a free type, available for service to put any element in it to give extra information on the encountered business errors returned by the target service.	No	No

StatusCode is recursive; therefore, *StatusCode* (level 1) could be embedded by an optional sub *StatusCode* (sub level). Each *StatusCode* must have a value attribute and there must be at least a level 1 *StatusCode*.

The response returns at least Level 1 *StatusCode* with one of the following values:

URI	Description
'urn:be:fgov:ehhealth:2.0:status:Success'	Completion of the request without errors.
'urn:be:fgov:ehhealth:2.0:status:Requester'	Completion of the request with errors caused by the WS consumer.
'urn:be:fgov:ehhealth:2.0:status:Responder'	Completion of the request with errors caused by the WS provider.

The optional Level 2 *StatusCode*, if returned, may have different values indicating specific cause of the error such as invalid input, missing input, data not found etc.

URI	Description
'urn:be:fgov:health:2.0:status:Intermediate'	Unknown error.
'urn:be:fgov:health:2.0:status:InvalidInput'	Invalid input error.
'urn:be:fgov:health:2.0:status:MissingInput'	Missing input.
'urn:be:fgov:health:2.0:status:DataNotFound'	No results for the request.
'urn:be:fgov:health:2.0:status:RequestDenied'	Unauthorized request (business level).
'urn:be:fgov:health:2.0:status:RequestUnsupported'	Service does not support the request.

Example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns8:GetSheet ... Offset="0" MaxElements="100" Id="_14cc837e-de41-4b38-b23a-f19a91148a83" InResponseTo="bb16782e9-9cea-4af4-8ce4-e1abe70a9687" IssueInstant="2016-04-07T10:40:57.881+02:00">
      <ns4:Status>
        <ns4:StatusCode Value="urn:be:fgov:health:2.0:status:Success">
          <ns4:StatusCode
Value="urn:be:fgov:health:2.0:status:InvalidInput"/>
        </ns4:StatusCode>
        <ns4:StatusMessage>KMEHR rule 22.3 validation
error.</ns4:StatusMessage>
      </ns4:Status>
    </ns8:GetSheet>
  </soap:Body>
</soap:Envelope>
```

See Chap 8 Error and failure messages for further description of *StatusCode* used in this service.

5.3.4 DeleteSheet

5.3.4.1 Request

The DeleteSheet request is executed using the RecordNumber of a sheet. It can be used on a sheet part A but also on a sheet with part A and B.

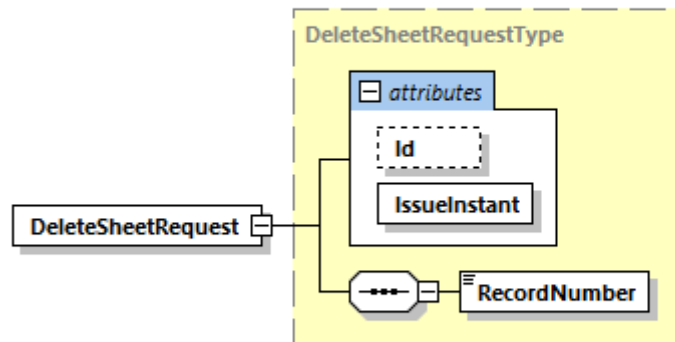


Figure 8: DeleteSheetRequest

Field Name	Description	Attribute	Required
Id	Identifier of the request within the caller system.	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
RecordNumber	Sheet number consisting of max 20 characters.	No	Yes

5.3.4.2 Response

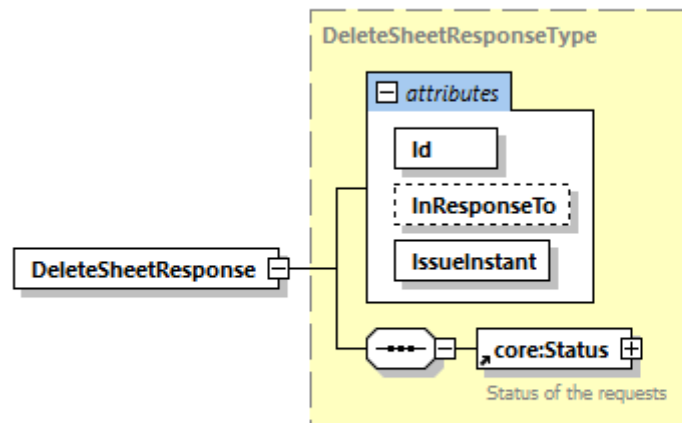


Figure 9: DeleteSheetResponse

Field Name	Description	Attribute	Required
Id	Identifier of the response.	Yes	Yes
InResponseTo	Id attribute of the request	Yes	No
IssueInstant	Date and time of the response.	Yes	Yes
Status	See section 5.4.3.1 StatusType	No	Yes

5.3.4.3 Example

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:ehhealth:emsr:registration:v1">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:DeleteSheetRequest Id="id-FBFE19453D50A4BFE61491476862891174" IssueInstant="2020-07-
31T09:18:48.288+02:00">
      <urn:RecordNumber>U777RPUM0GUQ6</urn:RecordNumber>
    </urn:DeleteSheetRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns4:DeleteSheetResponse Id="_5f9f8671-d4d9-4a0f-9e88-6d26d1b29482" InResponseTo="id-
FBFE19453D50A4BFE61491476862891174" IssueInstant="2020-07-31T09:18:20.875+02:00">
```

```

xmlns:ns10="urn:be:fgov:health:monitoring:protocol:v2"
xmlns:ns2="http://www.w3.org/2000/09/xmldsig#" xmlns:ns3="http://www.w3.org/2001/04/xmlenc#"
xmlns:ns4="urn:be:fgov:health:emsr:registration:v1" xmlns:ns5="urn:be:fgov:health:commons:core:v2"
xmlns:ns6="urn:be:fgov:health:commons:protocol:v2"
xmlns:ns7="http://www.ehealth.fgov.be/standards/kmehr/schema/v1"
xmlns:ns8="urn:be:fgov:health:emsr:core:v1" xmlns:ns9="urn:be:fgov:health:errors:service:v1">
  <ns5:Status>
    <ns5:StatusCode Value="urn:be:fgov:health:2.0:status:Success"/>
  </ns5:Status>
</ns4:DeleteSheetResponse>
</soap:Body>
</soap:Envelope>

```

5.3.5 AnonymizePatientData

The AnonymizePatientData request erases the identity information of a patient from a sheet.

The following information are erased from the sheet (for more information about the fields, please check external document 4, "Lijst met de AMBUREG-variabelen") :

- 29.1 patient.patientNationalNumber
- 29.2 patientName
- 30 patientFirstname
- 31.1 patient.patientInterventionAddrFlag
- 31.4 patient.patientAddressAbroadFlag
- 31.4.1 patient.patientAddressCountryCode
- 31.5 patient.patientZipCode
- 31.6 patient.patientLocality
- 31.8 patient.patientStreet
- 31.9 patient.patientStreetNo
- 31.10 patient.patientbox
- 31.10.1 patientPatientForeignAdress/FreeText
- 31.11 invoicingaddresspatientaddressFlag
- 31.12 invoicing.name
- 31.13 invoicing.Firstname
- 31.14 invoicing.Street
- 31.15 invoicing.Streetno
- 31.16 invoicing.Zipcode
- 31.17 invoicing.Locality
- 31.18 invoicing.box
- 31.19 invoicing.AbroadFlag
- 31.19.1 invoicing.CountryCode
- 31.20 invoicing.abroadFreeText

- 32.1 patient.patientDob
- 33 patient.patientSexCode

After the execution of AnonymizePatientData, no further RequestPartB or getSheet for that sheetnumber will be accepted.

5.3.5.1 Request

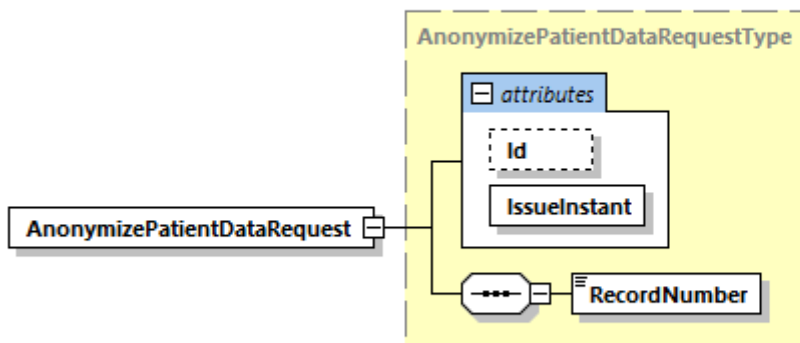


Figure 10: AnonymizePatientDataRequest

Field Name	Description	Attribute	Required
Id	Identifier of the request within the caller system.	Yes	No
IssueInstant	Date and time of the request.	Yes	Yes
RecordNumber	Sheet number consisting of a max of 20 characters.	No	Yes

5.3.5.2 Response

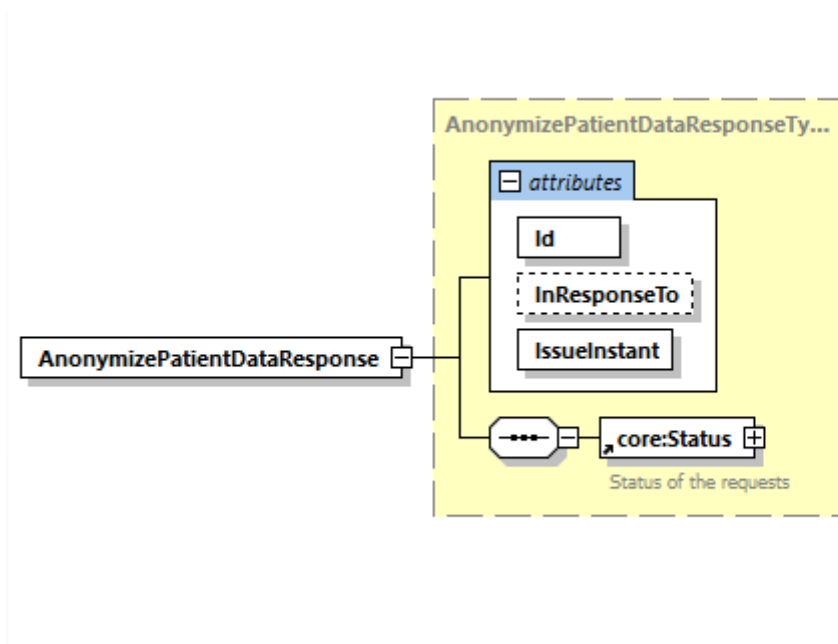


Figure 11: AnonymizePatientdataResponse

Field Name	Description	Attribute	Required
Id	Identifier of the response.	Yes	Yes
InResponseTo	Id attribute of the request	Yes	No
IssueInstant	Date and time of the response.	Yes	Yes
Status	See section 5.4.3.1 StatusType	No	Yes

5.3.5.3 Example

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:health:emsr:registration:v1">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:AnonymizePatientDataRequest Id="id-FBFE19453D50A4BFE61491476862891174"
IssueInstant="2020-07-31T10:58:48.288+02:00">
      <urn:RecordNumber>YYOUAGV7ENZKJ</urn:RecordNumber>
    </urn:AnonymizePatientDataRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:be:fgov:health:emsr:registration:v1">
  <soapenv:Header/>
```

```
<soapenv:Body>  
  <urn:AnonymizePatientDataRequest Id="id-FBFE19453D50A4BFE61491476862891174"  
IssueInstant="2020-07-31T10:58:48.288+02:00">  
  <urn:RecordNumber>YYOUAGV7ENZKJ</urn:RecordNumber>  
  </urn:AnonymizePatientDataRequest>  
</soapenv:Body>  
</soapenv:Envelope>
```

6. Risks and security

6.1 Security

6.1.1 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.

In case eHealth finds a bug or vulnerability in its software, the partner is advised 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 WS that eHealth delivered, he is obliged to contact and inform eHealth immediately and he is not allowed to publish this bug or vulnerability in any case.

6.1.2 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.
- Signature of the timestamp, body and binary security token. This will allow eHealth to verify the integrity of the message and the identity of the message author.

6.1.3 The use of username, password and token

The username, password, and token are strictly personal and are not allowed to transfer.

Every user takes care of his username, password, and token and is forced to confidentiality of it. Every user is also responsible of every use, which includes the use by a third party, until the inactivation.



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, 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 WS. Most of the required integration info to integrate is published 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 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 eHealth 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 eHealth 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 its applications will always test first in the acceptance environment before releasing any adaptations of its application in production. In addition, he will inform eHealth on the progress and test period.

7.2 Test cases

The eHealth platform recommends performing tests for all of the following cases:

- Publish multiple different part A and part B, and check correct handling of error message received.
- Search for one or more SDS.
- Delete a sheet.
- Anonymize a sheet.

In addition, the organization should also run negative test cases:

- Publish a part A or part B already present.
- Publish part B when part A is not already there.
- Delete a sheet that does not exist or has already been deleted.



- Anonymize a sheet that already has been anonymized.



8. Error and failure messages

There are three different possible types of response:

- If there are no technical or business errors, business response is returned.
- If a business error occurred, a business error is returned (see chapter 8.1 Business errors).
- In the case of a technical error, a SOAP fault exception is returned (see chapter 0).

8.1 Business errors

8.1.1 Status Code

See 5.4.4.1 StatusType for description of the StatusCode mechanism.

Business errors are forwarded without any transformation (they are treated as regular business responses).

These error codes first indicate a problem in the arguments sent.

StatusCode	Message	Solution
urn:be:fgov:ehhealth:2.0:status:Success (level 1) urn:be:fgov:ehhealth:2.0:status:DataNotFound (level 2)	No results for the request	Change one of the search criteria. Riduce the number of search criteria.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:InvalidInput (level 2)	KMEHR message validation error	Correct data in sheet.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	KMEHR part A does not exist.	First send part A before part B
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	KMEHR part A already exists.	Part A cannot be sent again
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	KMEHR part B already exists.	Part B cannot be sent again
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestUnsupported (level 2)	KMEHR message decryption error	The transmitted data cannot be decrypted
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestUnsupported (level 2)	KMEHR message XSD validation error	The transmitted data cannot be deserialized
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:InvalidInput (level 2)	The Offset attribute cannot be negative.	Offset must be ≥ 0 . It can be higher than MaxElements
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:InvalidInput (level 2)	The MaxElements attribute is too high.	MaxElements may not exceed 100 items per request
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:InvalidInput (level 2)	The MaxElements	MaxElements must be ≥ 0



	attribute cannot be negative or zero.	
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	Combination of team identification and intervention number does not exist in SDS-database.	There is no SDS sheet matching the search criteria entered.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:InvalidInput (level 2)	The content of the Base64Encrypted Data cannot be decrypted.	The encrypted part of the request (i.e. the folder part of the KMEHR) could not be decrypted because it is not encrypted correctly for EMSR (see section 3.1).
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	The sheet is already anonymized.	The sheet has already been anonymized and cannot be anonymized a second time.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:RequestDenied (level 2)	The sheet is not yet exported and cannot be anonymized.	To anonymize a sheet, it has to be already exported in the daily SPF Health export. Wait 24h before redoing the request.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:DataNotFound (level 2)	Sheet not found	The RecordNumber is not correct. Please check the RecordNumber.
urn:be:fgov:ehhealth:2.0:status:Requester (level 1) urn:be:fgov:ehhealth:2.0:status:Forbidden (level 2)	The team is not a member of your service.	Check the TeamIdentification in the request, it does not make part of the ambulance service.

8.1.2 Status Detail

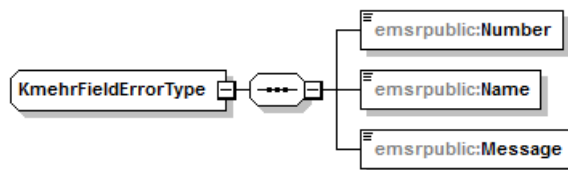


Figure 12: StatusDetail structure

If there was a validation error with the encrypted KMEHR content, a set of 'KmehrFieldType's will be returned. Each one corresponds to one rule violation with an explanation as described below.

Field Name	Description	Attribute	Required
Number	Rules number as in the excel sheet 'ambureg_variables' that is being violated.	No	Yes
Name	Enunciation of the failing rule	No	Yes
Message	This describes why the rule fails	No	Yes



Example of rule validation errors:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <emsr:RegisterPartAResponse Id="_de2754ca-83fe-41ce-9c72-9c3a7f586b38" InResponseTo="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3" IssueInstant="2016-04-07T10:09:48.288+02:00" ...>
      <core:Status>
        <core:StatusCode>
          Value="urn:be:fgov:health:2.0:status:Requestor">
            <core:StatusCode>
              Value="urn:be:fgov:health:2.0:status:InvalidInput"/>
            </core:StatusCode>
            <core:StatusMessage>KMEHR message validation error</core:StatusMessage>
            <core:StatusDetail>
              <emsrregister:KmehrFieldError>
                <Number>4.1</Number>
                <Name>missionIdentification</Name>
                <Message>the value should match
                "^\d{11}$"</Message>
              </emsrregister:KmehrFieldError>
              <emsrregister:KmehrFieldError>
                <Number>4.2</Number>
                <Name>interHospitalTransportFlag</Name>
                <Message>the value must be a
                boolean</Message>
              </emsrregister:KmehrFieldError>
            </core:StatusDetail>
          </core:Status>
        </emsr:RegisterPartAResponse>
      </soap:Body>
    </soap:Envelope>
```

Example KMEHR message not compliant to KMEHR XSD:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <emsr:RegisterPartAResponse Id="_de2754ca-83fe-41ce-9c72-9c3a7f586b38" InResponseTo="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3" IssueInstant="2016-04-07T10:09:48.288+02:00" ...>
      <core:Status>
        <core:StatusCode>
          Value="urn:be:fgov:health:2.0:status:Requestor">
            <core:StatusCode>
              Value="urn:be:fgov:health:2.0:status:RequestUnsupported"/>
            </core:StatusCode>
            <core:StatusMessage>KMEHR message XSD validation error</core:StatusMessage>
          </core:Status>
        </emsr:RegisterPartAResponse>
      </soap:Body>
    </soap:Envelope>
```



Example part A missing when sending part B:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <emsr:RegisterPartAResponse Id="_de2754ca-83fe-41ce-9c72-9c3a7f586b38" InResponseTo="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3" IssueInstant="2016-04-07T10:09:48.288+02:00" ...>
      <core:Status>
        <core:StatusCode
Value="urn:be:fgov:health:2.0:status:Requestor">
          <core:StatusCode
Value="urn:be:fgov:health:2.0:status:RequestDenied"/>
        </core:StatusCode>
        <core:StatusMessage>KMEHR part A does not exist</core:StatusMessage>
      </core:Status>
    </emsr:RegisterPartAResponse>
  </soap:Body>
</soap:Envelope>
```

Example when part A already exists:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <emsr:RegisterPartAResponse Id="_de2754ca-83fe-41ce-9c72-9c3a7f586b38" InResponseTo="bdc38ae62-3e7f-4f80-80f7-c3e745500fa3" IssueInstant="2016-04-07T10:09:48.288+02:00" ...>
      <core:Status>
        <core:StatusCode
Value="urn:be:fgov:health:2.0:status:Requestor">
          <core:StatusCode
Value="urn:be:fgov:health:2.0:status:RequestDenied"/>
        </core:StatusCode>
        <core:StatusMessage>KMEHR part A already exists</core:StatusMessage>
      </core:Status>
    </emsr:RegisterPartAResponse>
  </soap:Body>
</soap:Envelope>
```

8.2 Technical errors

Technical errors are errors inherent to the internal working of a WS. They are returned as SOAP Faults. The SOA Standard for Errorhandling specifies a structure for SystemError and BusinessError, thrown as SOAP Faults.

A **SystemError** MUST be thrown when a system failure occurred. It is not related to the business of the service. The SOA system error structure is as follows:



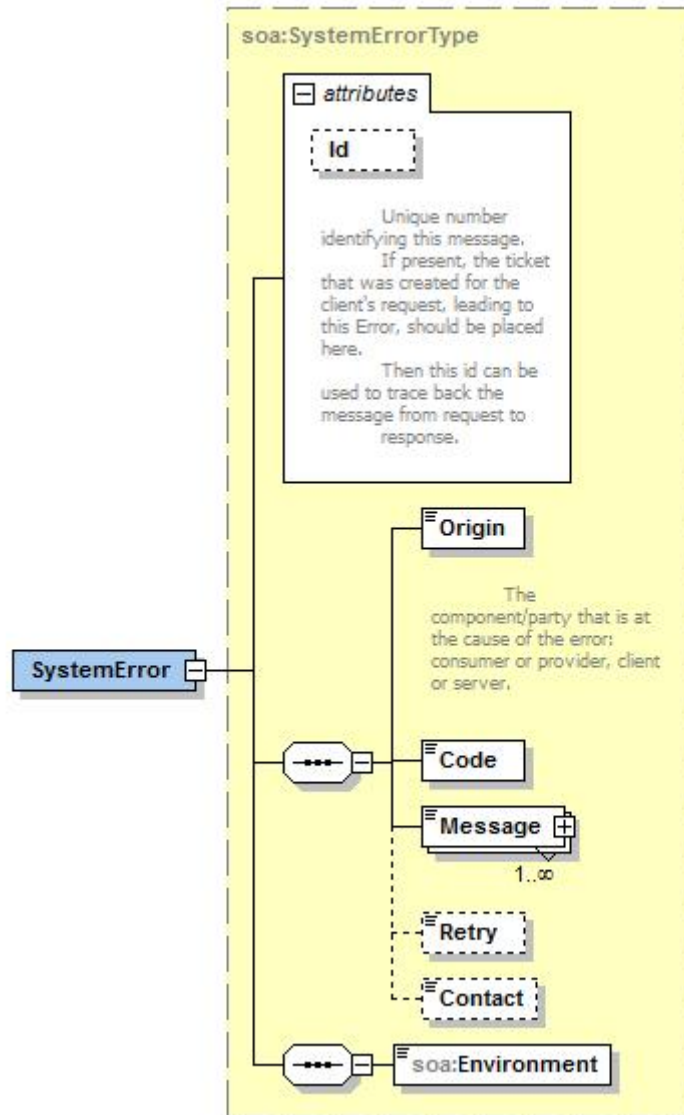


Figure 13 SystemError

The SystemError element MUST contain a unique Id attribute for tracing.

The Origin MUST be set to Server or Provider.

Retry SHOULD be set to true if the consumer can try again immediately without interventions.

Example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <soapenv:Fault>
      <faultcode>soapenv:Server</faultcode>
      <faultstring>SOA-02002</faultstring>
      <detail>
        <soa:SystemError Id="9E0-00000P1-00-C" xmlns:soa="urn:be:fgov:ehhealth:errors:soa:v1">
          <Origin>Server</Origin>
          <Code>SOA-02002</Code>
          <Message xml:lang="en">Service is temporarily not available. Please contact service
desk.</Message>
          <Retry>true</Retry>
          <soa:Environment>Test</soa:Environment>
        </soa:SystemError>
      </detail>
    </S:Fault>
  </soapenv:Body>
</soapenv:Envelope>
```

The SOAP Fault element has the following sub elements:

Element name	Descriptions	Required
faultcode	A code for identifying the fault	Yes
faultstring	A human readable explanation of the fault	Yes
faultactor	Information about who/what caused the fault to happen (the origin)	No
detail	Holds application specific error information related to the Body element. For example, it could include a java stack trace or any other kind of trace, used internally, to document on the cause of this error.	No

The default SOAP faultcode values are defined in an extensible manner that allows for new SOAP fault code values to be defined while maintaining backwards compatibility with existing fault code values.

Element name	Descriptions
versionMismatch	Found an invalid namespace for the SOAP Envelope element.
mustUnderstand	An immediate child element of the Header element, with the mustUnderstand attribute set to "1", was not understood.
client	Incorrectly formed message or containing incorrect information.
server	There was a problem with the server so the message could not proceed.

Description of the possible SOAP fault exceptions:

Error code	Component	Description	Solution/Explanation
SOA-00001	Undefined	Service error	This is the default error sent to the consumer in case more details are missing.



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	An unexpected error has occurred <ul style="list-style-type: none"> • 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 default error for content related errors in case no more details are known.
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.