eHealth Id Support WS
Cookbook

Version 1.5

This document is provided to you, free of charge, by the

eHealth platform

Willebroekkaai 38 – 1000 Brussel

38, Quai de Willebroek – 1000 Bruxelles

Table of contents

Table of Contents

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	14/04/2014	eHealth platform	First version of the document.
1.1	04/09/2014	eHealth platform	Additional end-user profile: Hospital Additional support card: Biometric residence permit
1.2	11/07/2017	eHealth platform	Additional end-user profile : Hub
1.3	23/04/2020	eHealth platform	WS-I Compliance
1.4	23/04/2021	eHealth platform	Update business errors
1.5	19/07/2022	eHealth platform	§ 2.3 eHealth platform document references (updated) § 3.2 Status (added) § 5.1.3 Tracing (updated)

2. Introduction

2.1 Goal of the service

The web service (WS) aims to support the verification of the validity of identification supports for physical persons in Belgium. It currently covers the following kinds of supports: eID, KidsID, ForeignID, biometric residence permit, SIS and ISI+ cards.

This service is intended to be used only by authorized healthcare professionals and institutions in the context of the care of the patients and the implied relations with the Belgian Social Security (e.g. insurance status verification).

The service requires as main inputs

EITHER

- A couple of data composed of the Belgian Identification Number for Social Security (INSS) of a person and the number of the support card number resuming this identification.

OR

- The two-dimensional bar code figuring on the e-ID or ISI+ card. This bar code covers the Belgian Identification Number for Social Security (INSS) of a person and the number of the support card.

The eHealth platform is certainly not the authentic source for data related to citizen's identities thus the service is in fact only a "relay" service to the service provided for this purpose by the Crossroads Bank of Social Security.

2.2 Goal of the document

In this cookbook, we explain the structure and content aspects of the possible requests and the replies of eHealth platform Id Support WS. An example illustrates each of those messages. In addition, you can find a list of possible errors 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 the eHealth platform WS in client applications.

This document is neither a development nor a programming guide for internal applications; eHealth platform 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 platform 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 Document references

All the document references can be found on the eHealth portal[1]. 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	STS Cookbook	1.5	13/07/2022	eHealth platform

[1] ***https://www.ehealth.fgov.be/ehealthplatform***

3	https://www.ehealth.fgov.be/ehealthplatform/ehealth_i.am_-_federation_attributes_v1.2_27042017.pdf	1.2	26/04/2017	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-SAMLTokenProfile.pdf***	01/02/2006	OASIS Standard
2.	Basic Profile Version 1.1	***http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html***	24/08/2004	Web Services Interoperability Organization

2.5 Service history

Version	Release date	Changes
1.0	Major release 2014.1 - Move to Acceptance: 31/03/2014 - Move to Production: 11/05/2014	The access of the eHealth platform WS Id Support v2 is granted to Nurse, Physician, Pharmacy, Dentist, and Physiotherapist.
1.1	Minor release 2014.1.2 - Move to Acceptance: 12/08/2014 - Move to Production: 28/08/2014	Grant access to Hospital.
1.2	Out-of-release - Move to Acceptance: 21/07/2017 - Move to Production: 31/07/2017	Grant access to Hub

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.

4. Global overview

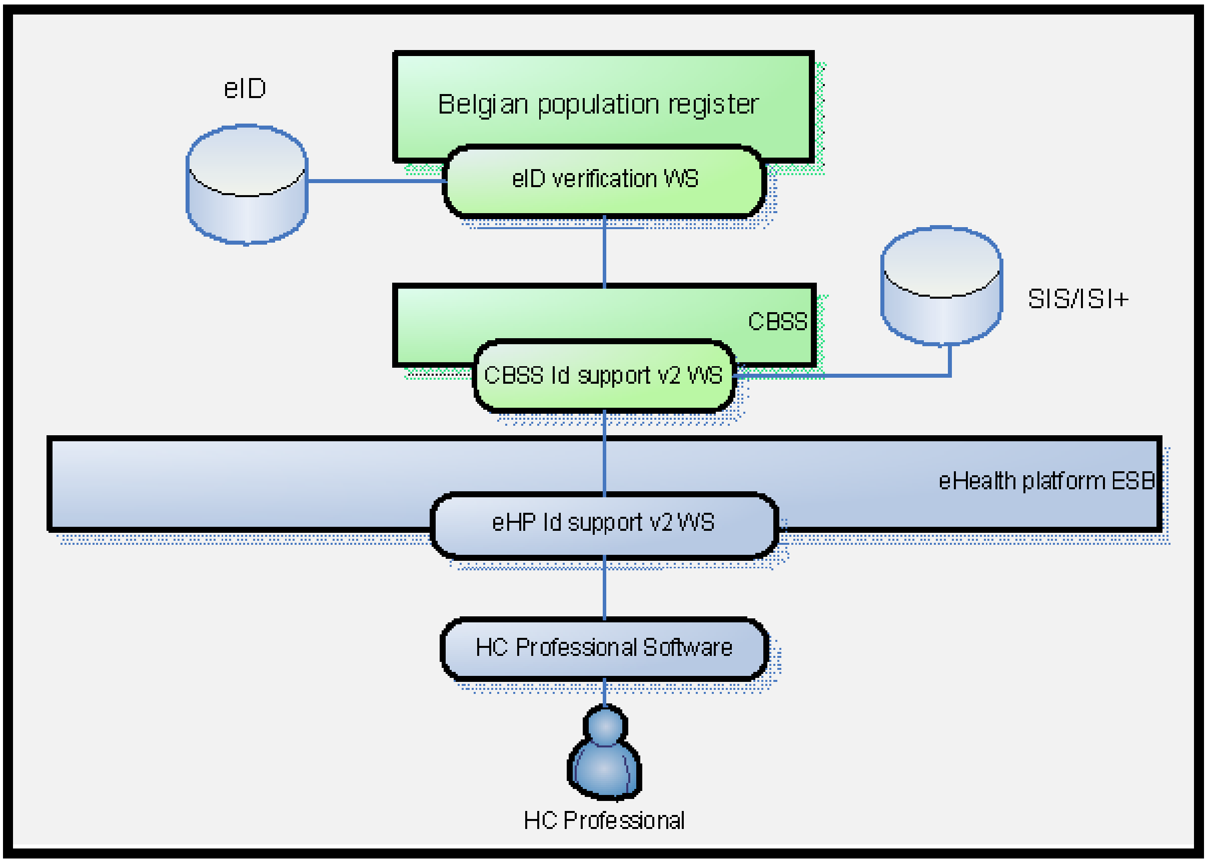


Figure 1: Overview of the eHealth Id support web service.

This service is considered as a relay service as the verification is made at the CBSS. Indeed, the CBSS has a database of the issued cards SIS/ISI+ and provides a service to check the status of Belgian identity cards stored in the Belgian Population Register database.

5. Step-by-step

5.1 Technical requirements

5.1.1 Use of the eHealth SSO solution

This section specifies how to obtain a SAML token from the Secure Token Service (STS) in order to have access to the eHealth platform Id Support WS. There are different types of user, according to eHealth platform's Unique File, who are allowed to access the eHealth platform Id Support WS and act as author of operation's requests, therefore this document will be updated when the services are made available to a new type of user. These different groups of user are described hereunder.

Each type of user needs a different type of token to access the service. The remainder of this section describes the needed attributes for each type of the user. For more details on how STS works, please refer to:

- Dutch version: ***https://www.ehealth.fgov.be/ehealthplatform/nl/service-iam-identity-access-management***
- French version : ***https://www.ehealth.fgov.be/ehealthplatform/fr/service-iam-identity-access-management***

5.1.2 WS-I Basic Profile 1.1

Your request must be WS-I compliant (See Chap 2.4 - External document references).

5.1.3 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***

5.1.3.1 Nurse

The request for the SAML token is secured with the eID[2] of the nurse. The certificate used by the Holder-Of-Key (HOK) verification mechanism is an eHealth platform certificate. The needed attributes are the following (AttributeNamespace: "urn:be:fgov:identification-namespace"):

- The social security identification number of the nurse:
    - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*

[2] *As fallback, in absence of the eID, the personal eHealth platform certificate can be used for authentication instead.*

- *urn:be:fgov:person:ssin*

Nurse must also specify which information must be asserted by eHealth:

- The social security identification number of the nurse: (AttributeNamespace: "urn:be:fgov:identification-namespace"):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*
- To have access to the Id Support WS, the person must be a nurse: (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*)
  - *urn:be:fgov:person:ssin:ehealth:1.0:fpsph:nurse:boolean*
- Nurse uses his/her personal certificate (AttributeNamespace: "urn:be:fgov:certified-namespace:ehealth")
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin:usersession:boolean*

5.1.3.2 Physician

The request for the SAML token is secured with the eID[3] of the doctor. The certificate used by the HOK verification mechanism is an eHealth platform certificate. The needed attributes are the following (AttributeNamespace: "urn:be:fgov:identification-namespace"):

- The social security identification number of the doctor:
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*

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

- The social security identification number of the doctor (AttributeNamespace: "urn:be:fgov:identification-namespace"):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*
- To have access to the eHealth platform Id Support web service, the person must be a doctor (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*):
  - *urn:be:fgov:person:ssin:ehealth:1.0:fpsph:doctor:boolean*
- Doctor uses his/her personal certificate (AttributeNamespace: "urn:be:fgov:certified-namespace:ehealth")
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin:usersession:boolean*

5.1.3.3 Pharmacy

Pharmacies must specify several attributes in the request. The request to the STS is secured with the eID of the pharmacist starting the session. The certificate of the pharmacy issued by the eHealth platform is used by the HOK mechanism. The attributes that need to be provided in the request are the following (AttributeNamespace: *urn:be:fgov:identification-namespace*):

- The social security identification number of the person starting the session (must be a pharmacist):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*

[3] *As fallback, in absence of the eID, the personal eHealth platform certificate can be used for authentication instead.*

- The identification of the pharmacy:
  - *urn:be:fgov:ehealth:1.0:pharmacy:nihii-number*
- The identification of the pharmacy holder:
  - *urn:be:fgov:person:ssin:ehealth:1.0:pharmacy-holder*

Pharmacies must also specify which information must be asserted by the eHealth platform. To have access to the eHealth platform Id Support WS, the following data must be validated:

- The SSIN of the person (must be a pharmacist) starting the session, this is verified by the eHealth platform (AttributeNamespace: *urn:be:fgov:identification-namespace*):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*
- The NIHII number of the pharmacy. The link between the pharmacy and the pharmacist starting the session is not verified, any pharmacist can start the session (AttributeNamespace: *urn:be:fgov:identification-namespace*):
  - *urn:be:fgov:ehealth:1.0:pharmacy:nihii-number*
- The pharmacy must be a recognized pharmacy (AttributeNamespace: *urn:be:fgov:certified-namespace:ehealth*):
  - *urn:be:fgov:ehealth:1.0:pharmacy:nihii-number:recognisedpharmacy:boolean*
- The identification of the pharmacy holder (SSIN), i.e. the pharmacist responsible for all activities performed in the pharmacy (AttributeNamespace: *urn:be:fgov:identification-namespace*):
  - *urn:be:fgov:person:ssin:ehealth:1.0:pharmacy-holder*
- The identification of the pharmacy holder (NIHII11), i.e. the pharmacist responsible for all activities performed in the pharmacy (AttributeNamespace: *urn:be:fgov:certified-namespace:ehealth*):
  - *urn:be:fgov:person:ssin:ehealth:1.0:pharmacy-holder:certified:nihii11*
- The pharmacy holder must be the certified pharmacy holder of the given pharmacy (AttributeNamespace: urn:be:fgov:certified-namespace:ehealth):
  - *urn:be:fgov:ehealth:1.0:pharmacy:nihii-number:person:ssin:ehealth:1.0:pharmacy-holder:boolean*
- To have access to the eHealth platform Id Support web service, the person must be a pharmacist (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*):
  - *urn:be:fgov:person:ssin:ehealth:1.0:fpsph:pharmacist:boolean*
- Pharmacist uses his/her personal certificate (AttributeNamespace: "urn:be:fgov:certified-namespace:ehealth")
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin:usersession:boolean*

5.1.3.4 Dentist

The request for the SAML token is secured with the eID[4] of the dentist. The certificate used by the HOK verification mechanism is an eHealth platform certificate. The needed attributes are the following (AttributeNamespace: "urn:be:fgov:identification-namespace"):

- The social security identification number of the dentist:
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*

[4] *As fallback, in absence of the eID, the personal eHealth certificate can be used for authentication instead.*

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

- The social security identification number of the dentist (AttributeNamespace: "urn:be:fgov:identification-namespace"):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*
- To have access to the eHealth platform Id Support WS, the person must be a dentist (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*):
  - *urn:be:fgov:person:ssin:ehealth:1.0:fpsph:dentist:boolean*
- Dentist uses his/her personal certificate (AttributeNamespace: "urn:be:fgov:certified-namespace:ehealth")
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin:usersession:boolean*

5.1.3.5 Physiotherapist

The request for the SAML token is secured with the eID[5] of the physiotherapist. The certificate used by the HOK verification mechanism is an eHealth platform certificate. The needed attributes are the following (AttributeNamespace: "urn:be:fgov:identification-namespace"):

- The social security identification number of the physiotherapist:
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*

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

- The social security identification number of the physiotherapist (AttributeNamespace: "urn:be:fgov:identification-namespace"):
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin*
  - *urn:be:fgov:person:ssin*
- To have access to the eHealth platform Support Id Support WS, the person must be a physiotherapist (AttributeNamespace: *urn:be:fgov:certifiednamespace:ehealth*)
  - *urn:be:fgov:person:ssin:ehealth:1.0:fpsph:physiotherapist:boolean*
- Physiotherapist uses his/her personal certificate (AttributeNamespace: "urn:be:fgov:certified-namespace:ehealth")
  - *urn:be:fgov:ehealth:1.0:certificateholder:person:ssin:usersession:boolean*

5.1.3.6 Hospital

The SAML token request is secured with the eHealth platform certificate of the hospital. The certificate used by the HOK verification mechanism is an eHealth platform certificate. The needed attributes are the following (Attribute namespace: "urn:be:fgov:identification-namespace"):

- The NIHII number of the hospital:
  - *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number*
  - *urn:be:fgov:ehealth:1.0:hospital:nihii-number*

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

- The NIHII number as identifier of the hospital (Attribute namespace: urn:be:fgov:identification-namespace):

  - *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number*
  - *urn:be:fgov:ehealth:1.0:hospital:nihii-number*

- To have access to the eHealth platform Consent WS, the hospital must be a recognized hospital (AttributeNamespace: urn:be:fgov:certified-namespace:ehealth):
  - *urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number:recognisedhospital:boolean*
  - *urn:be:fgov:ehealth:1.0:hospital:nihii-number:recognisedhospital:boolean*

5.1.3.7 Hub

Prior to calling the service, a SAML token proving that the call comes from a certified hub must be obtained from the eHealth platform TS Web Service.

In order to receive a token from the STS several attributes must be specified in the request. The attributes that need to be provided are the following:

- the EHP number as identifier of the Hub: AttributeNamespace="urn:be:fgov:identification-namespace",
  - *urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number*

Hub has also to specify which information must be validated by eHealth. To have access, the following data must be validated:

- the EHP number as identifier of the Hub (in two separate attributes): AttributeNamespace="urn:be:fgov:identification-namespace",
  - *urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number*
- Hub must be a recognized hub (AttributeNamespace="urn:be:fgov:certified-namespace:ehealth"),
  - *urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number:recognisedhub:boolean*

5.1.4 Encryption

Encryption (ETEE) is not used in the context of this project.

5.1.5 Security policies to apply

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

WS security used in this manner is in accordance with the common standards. To call the eHealth Id Support WS:

- Add the business message to the SOAPBody.
- Add to the SOAP header the following elements:
  - **SAML Token**: The SAML assertion received from the eHealth platform 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). (Please refer to 'List of source' chapter)
  - **Timestamp** (with Time-to-live of the message: one minute).
  - 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 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 WSDL[6].

For more information on the SAML token as needed to call the eHealth platform Id Support service (e.g., attributes needed by different actors), please refer to STS cookbook.

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

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

5.2 Web service

The Id Support Service has one operation ***verifyId*** to verify the status of the couple 'INSS' and the 'identification support' of a person.

This section describes the structure of the request and the response messages.

5.2.1 eHealth SOA standard structure

This service is 'eHealth SOA standards-based' WS. In particular, the following generic elements are essential to build the request and the reply of eHealth platform Id Support WS.

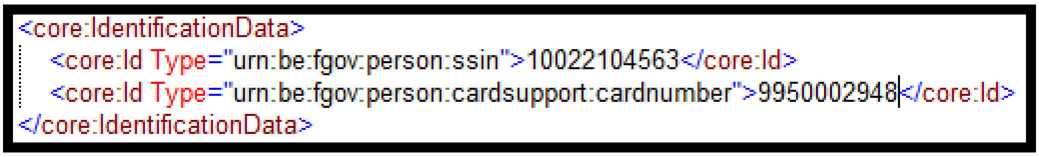
5.2.1.1 Id

This element offers the possibility for relying parties to send and receive data about users. These data, called user attributes or just attributes, can be anything that the provider knows about the user and that may be helpful to the service provider. Some of examples of this type of data are:

- The name of the user.
- The INSS of the user.
- The support card number of user's INSS.

The attributes are described in the paragraph 5.2.2. Additional information and attributes are described in the document I.AM Federation Attributes [IAM-DOC-ATTR].

Example:



```
<core:IdentificationData>
    <core:Id Type="urn:be:fgov:person:ssin">10022104563</core:Id>
    <core:Id Type="urn:be:fgov:person:cardsupport:cardnumber">9950002948</core:Id>
</core:IdentificationData>
```

5.2.1.2 Status

eHealth platform 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.

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:

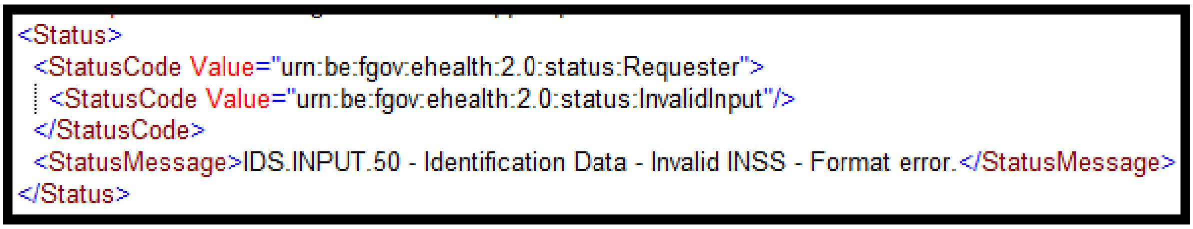
[6] *WSDL's can be found through* ***https://portal.api.ehealth.fgov.be/***

URI	Description
'urn:be:fgov:ehealth:2.0:status:Success'	Completion of the request without errors.
'urn:be:fgov:ehealth:2.0:status:Requester'	Completion of the request with errors caused by the WS consumer.
'urn:be:fgov:ehealth: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:ehealth:2.0:status:Intermediate'	Unknown error.
'urn:be:fgov:ehealth:2.0:status:InvalidInput'	Invalid input error.
'urn:be:fgov:ehealth:2.0:statusMissingInput'	Missing input.
'urn:be:fgov:ehealth:2.0:status:DataNotFound'	No results for the request.
'urn:be:fgov:ehealth:2.0:status:RequestDenied'	Unauthorized request (business level).
'urn:be:fgov:ehealth:2.0:status:RequestUnsupported'	Service does not support the request.

Example:



```
<Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Requester">
    <StatusCode Value="urn:be:fgov:ehealth:2.0:status:InvalidInput"/>
  </StatusCode>
  <StatusMessage>IDS.INPUT.50 - Identification Data - Invalid INSS - Format error.</StatusMessage>
</Status>
```

5.2.2 Method *verifyId*

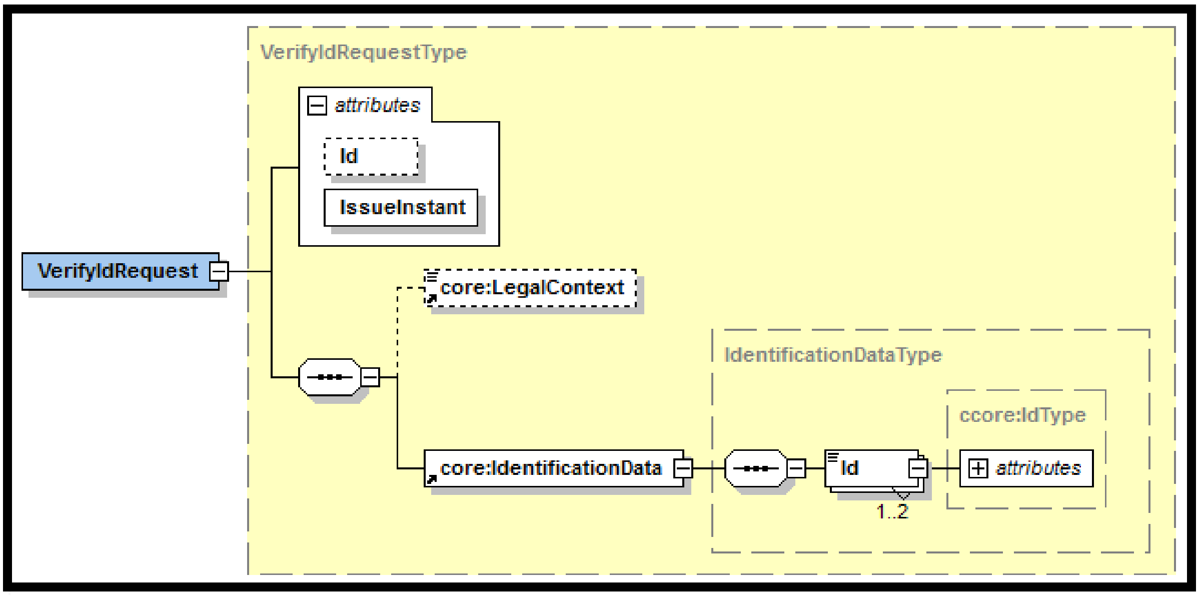
5.2.2.1 Functional description

Service name	verifyId
Purpose	The service aims to support the verification of the validity of identification supports for physical persons in Belgium.
Input parameters	1) The information about the request: • The identification of the request (optional, if provided then must be unique). • The issue instant: date and time of the request (mandatory). 2) The context of the verification (mandatory). The only supported value at this time is: "patient insurance validation ". 3) The identification data is the identity submitted for verification. Two options are offered by the service to verify the identity of a person through: • Option 1: INSS and support card number The identification data is composed of the INSS (mandatory) and the support card number (mandatory). The format of INNS and card number are as follows:

	○ INSS: 11 digits of the form *yymmddxxxcd* where *yymmdd* is the birthdate of the person, *xxx* a sequential number (odd for males and even for females) and *cd* a check-digit. Example: 84091304237 ○ Support card number: ▪ 12 digits for e-ID, Kids-card. Example: 591112548495 ▪ 9 digits preceded by letter ‘B’ for the foreign card. Example: B025275772 ▪ 9 digits for the biometric residence permit. Example: 123456789 ▪ 10 digits for the SIS, ISI+ card. Example: 1261804187 • Option 2: bar code The identification data is composed of a bar code covering both INSS and support card number (mandatory): ○ 20 digits for e-ID bar code. Example: 56052943602046558208 ○ 22 digits for ISI+ bar code. Example: 0705231386209951170180
Output parameters	1) The information about the response: • Identification of the response (mandatory, unique). • The identification of the request (mandatory if the request contains an id). • The issue instant of the response: date and time (mandatory). 2) An acknowledge indicating the completion of the request: • The status of the completion (mandatory). • Optionally, the result of the validation when the identification data are found in the DB with a negative status e.g. the person is dead, the support card is declared stolen or lost, the wrong combination of the INSS and card number, the card is expired or already destroyed.
Post-condition	The request is logged.
Possible exceptions	1) Technical error. 2) Business error: • Invalid or missing INSS. • Invalid or missing card number. • Invalid bar code. • Invalid identification data (combination of provided information). • There is no result for the provided data. • The data is not found in DB. • Invalid INSS and card number combination. • The card owner is dead. • The card is revoked. • The card is expired. • The card is destroyed.

	• The card is declared as stolen. • The card is declared as lost. • The card is not activated by the municipality.

5.2.2.2 *Input argument 'VerifyIdRequest'*



5.2.2.2.1 VerifyIdRequest

- **VerifyIdRequestType**

Define the '**VerifyIdRequest**' by gathering following information:

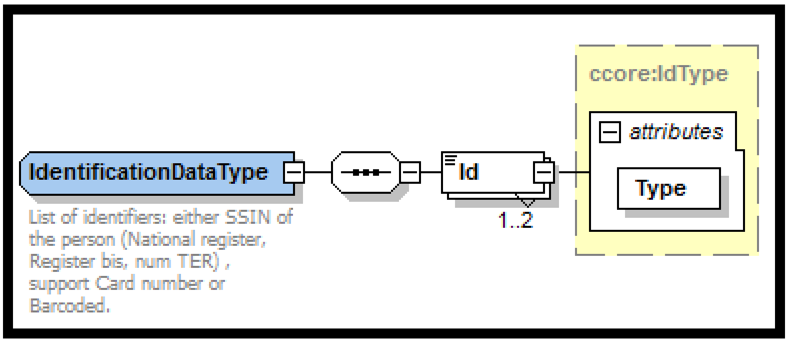
- The information about the request.
- The legal context of the request.
- The data submitted for verification.

VerifyIdRequestType			
Element	**Attributes**		**Comments**
	Id	Identifier of the request within the caller system.	Identifies the **message** within the system. Optional, if the Id is provided then this information must be unique. This Id is used for tracing the request.
	IssueInstant	Date and time of the request.	Mandatory. Format YYYY-MM-DDThh:mm:ssZ
LegalContext [0]		The context of the verification.	Mandatory. Currently, only supported and default value is ‘patient insurance validation’. This value is case sensitive, with space separating words.
IdentificationData [1]		Information submitted for verification.	Mandatory.

- **IdentificationDataType**

  Define The ***IdentificationData,*** composing of the Id element. The type of the Id indicates the provided information[7] :

  ○ The information submitted for verification (either INSS AND support card number OR bar code of e-ID, ISI+).


ccore:IdType
attributes
Type
IdentificationDataType
List of identifiers: either SSIN of the person (National register, Register bis, num TER) , support Card number or Barcoded.
Id
1..2

[7] *Please refer to the paragraph 5.2.1.2*

IdentificationDataType			
Element	**Attribute**	**Possible values**	**Comments**
Id [1-2]	Type	urn:be:fgov:person:ssin	INSS of the person. Mandatory, if the card number is provided and the bar code is not provided.
		urn:be:fgov:person:cardsupport:cardnumber	Support card number. Mandatory, if the SSIN is provided and the bar code is not provided.
		urn:be:fgov:person:cardsupport:barcoded	Support card bar code. Mandatory only if the SSIN and card number are not provided.

The Id support V2 distinguishes different support card through its format. Id Attributes used are described in the following page.

INSS

URI	**urn:be:fgov:person:ssin**
SAML 1.1 Namespace	identity
Xsi type	xs:string
Description	Identification Number of Social Security: 11 digits
Example	84091304237

Support card number

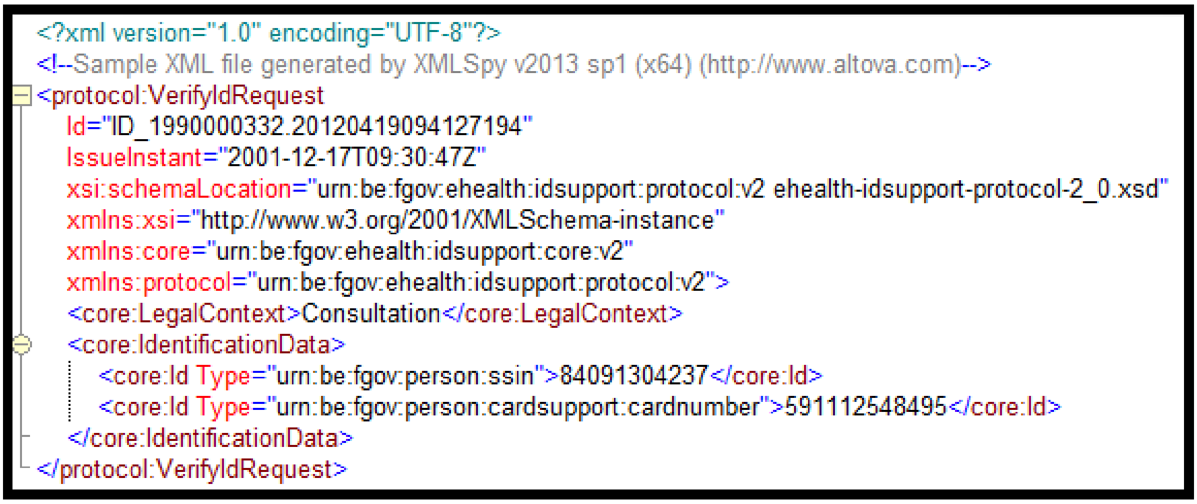
URI	**urn:be:fgov:person:cardsupport:cardnumber**
SAML 1.1 Namespace	Identity
Xsi type	xs:string
Description	Support card number format - eID or KidsCard : 12 digits - Foreign card: 9 digits preceded by the letter 'B' designating the type of card (A,B,C or D card). - Biometric residence permit: 9 digits. - SIS, ISI+: 10 digits
Example	e-ID : 591112548495 SIS : 1261804187 ISI+: 9951170180 Biometric residence permit: 123456789 Foreign card : B025275772

Barcode

URI	**urn:be:fgov:person:cardsupport:barcoded**
SAML 1.1 Namespace	identity
Xsi type	xs:string
Description	Barcoded of card support (eID, ISI+) - 20 digits for eID or KidsCard - 22 digits for ISI+
Example	eID: 77052628604006054038 ISI+: 0403011505109950001130

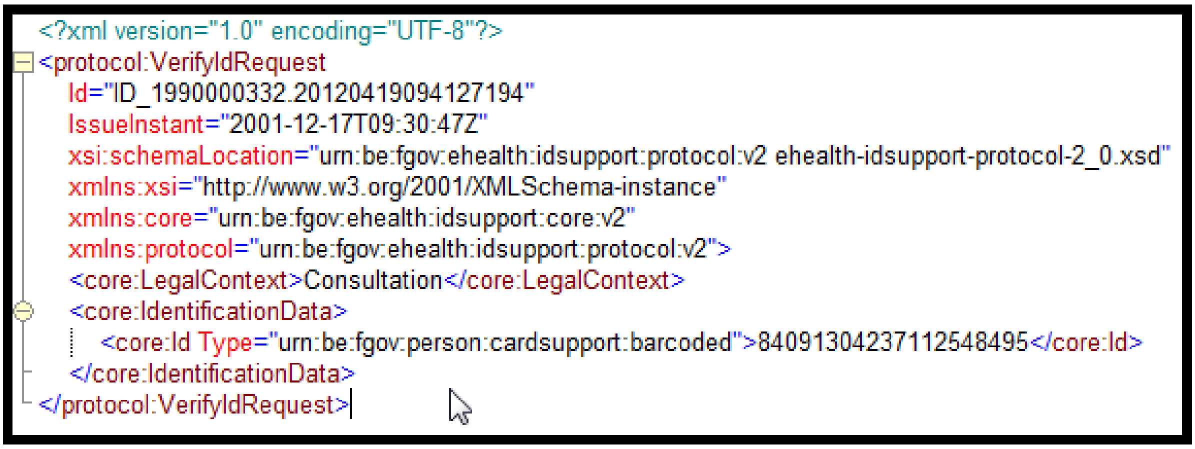
5.2.2.2.2 Examples of VerifyIdRequest.xml

- **Correct request 1 -** Verifying person identity with provided INSS and support card number



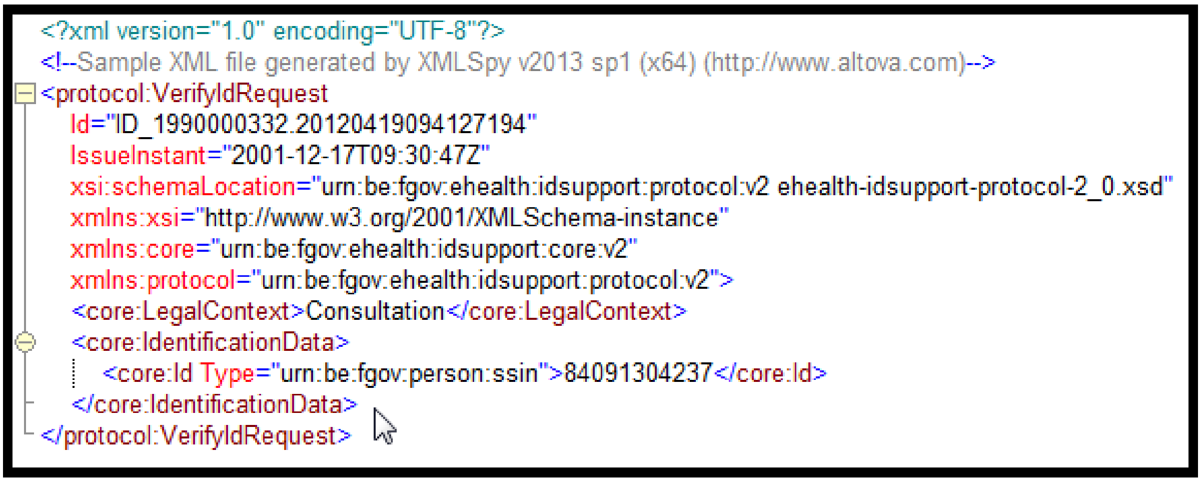
```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2013 sp1 (x64) (http://www.altova.com)-->
<protocol:VerifyIdRequest
    Id="ID_1990000332.20120419094127194"
    IssueInstant="2001-12-17T09:30:47Z"
    xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ehealth-idsupport-protocol-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
    xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
    <core:LegalContext>Consultation</core:LegalContext>
    <core:IdentificationData>
        <core:Id Type="urn:be:fgov:person:ssin">84091304237</core:Id>
        <core:Id Type="urn:be:fgov:person:cardsupport:cardnumber">591112548495</core:Id>
    </core:IdentificationData>
</protocol:VerifyIdRequest>
```

- **Correct request 2 -** Verifying person identity with provided support card bar code



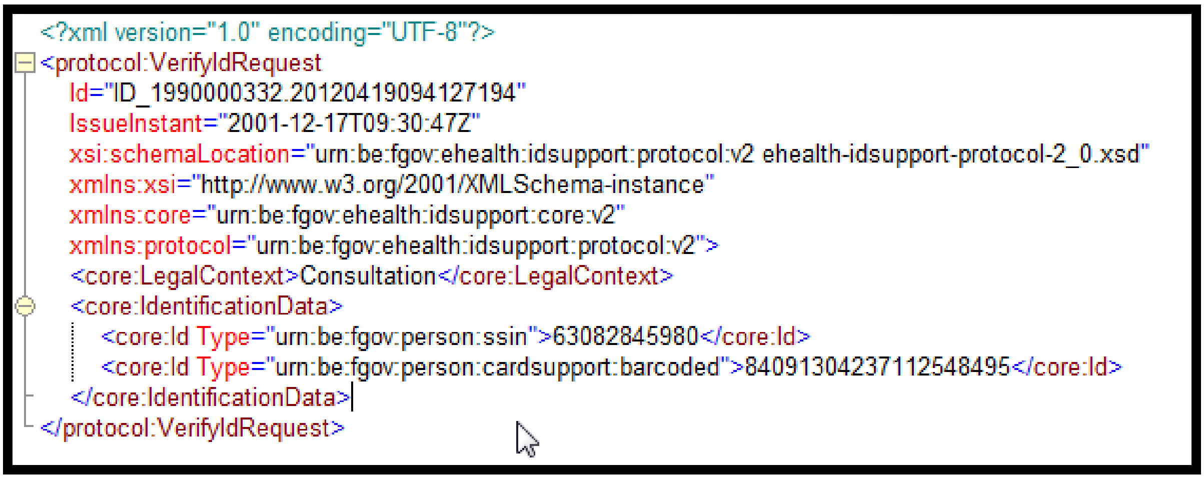
```
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdRequest
    Id="ID_1990000332.20120419094127194"
    IssueInstant="2001-12-17T09:30:47Z"
    xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ehealth-idsupport-protocol-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
    xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
    <core:LegalContext>Consultation</core:LegalContext>
    <core:IdentificationData>
        <core:Id Type="urn:be:fgov:person:cardsupport:barcoded">84091304237112548495</core:Id>
    </core:IdentificationData>
</protocol:VerifyIdRequest>
```

- **Incorrect request 1 –** Missing card number



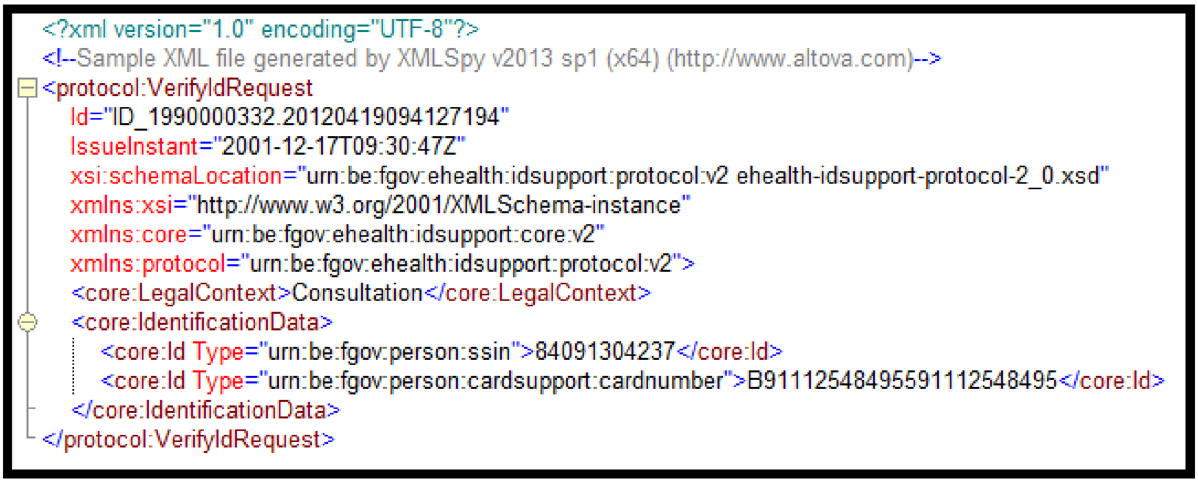
```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2013 sp1 (x64) (http://www.altova.com)-->
<protocol:VerifyIdRequest
    Id="ID_1990000332.20120419094127194"
    IssueInstant="2001-12-17T09:30:47Z"
    xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ehealth-idsupport-protocol-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
    xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
    <core:LegalContext>Consultation</core:LegalContext>
    <core:IdentificationData>
        <core:Id Type="urn:be:fgov:person:ssin">84091304237</core:Id>
    </core:IdentificationData>
</protocol:VerifyIdRequest>
```

- **Incorrect request 2 –** Incorrect combination INSS v/s bar code



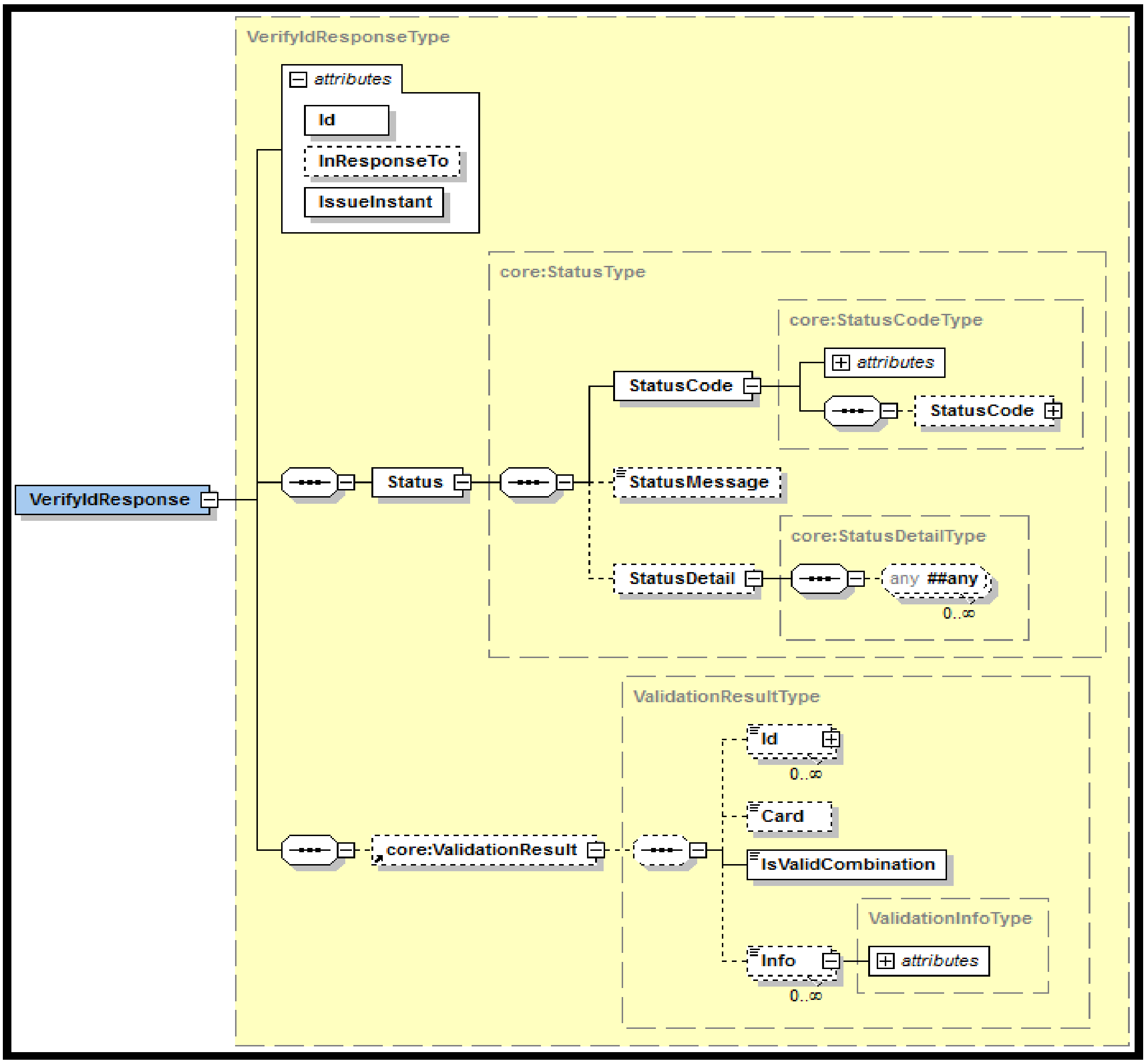
```xml
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdRequest
    Id="ID_1990000332.20120419094127194"
    IssueInstant="2001-12-17T09:30:47Z"
    xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ehealth-idsupport-protocol-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
    xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
    <core:LegalContext>Consultation</core:LegalContext>
    <core:IdentificationData>
        <core:Id Type="urn:be:fgov:person:ssin">63082845980</core:Id>
        <core:Id Type="urn:be:fgov:person:cardsupport:barcoded">84091304237112548495</core:Id>
    </core:IdentificationData>
</protocol:VerifyIdRequest>
```

- **Incorrect request 3 –** Invalid format of the support card's number



```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2013 sp1 (x64) (http://www.altova.com)-->
<protocol:VerifyIdRequest
    Id="ID_1990000332.20120419094127194"
    IssueInstant="2001-12-17T09:30:47Z"
    xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ehealth-idsupport-protocol-2_0.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
    xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
    <core:LegalContext>Consultation</core:LegalContext>
    <core:IdentificationData>
        <core:Id Type="urn:be:fgov:person:ssin">84091304237</core:Id>
        <core:Id Type="urn:be:fgov:person:cardsupport:cardnumber">B91112548495591112548495</core:Id>
    </core:IdentificationData>
</protocol:VerifyIdRequest>
```

5.2.2.3 *Output argument 'VerifyIdResponse'*

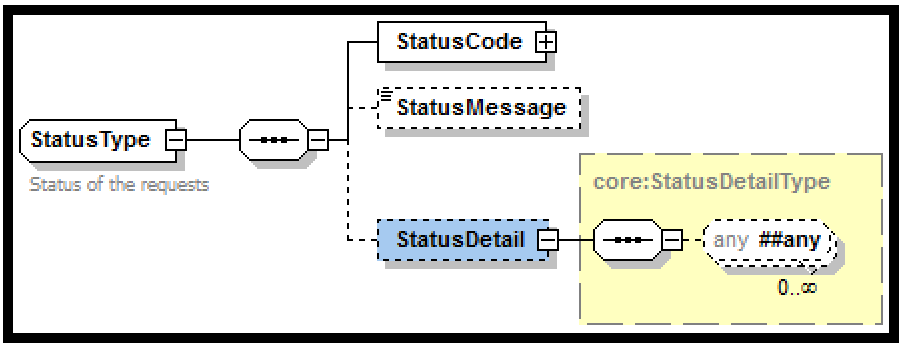


5.2.2.3.1 VerifyIdResponse

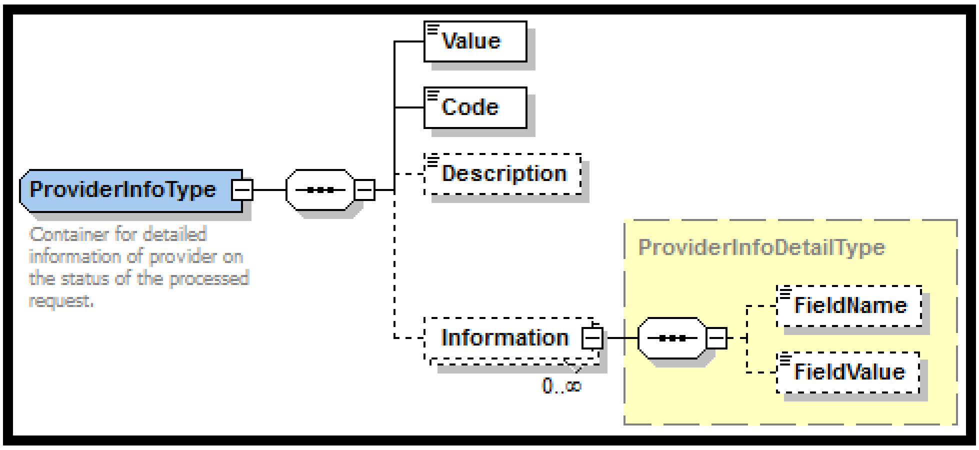
For additional information, please refer to the paragraph 5.2.1

- **StatusDetailType**

  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. In this case the target service returns information contained in the **ProviderInfo**



- **ProviderInfoType**

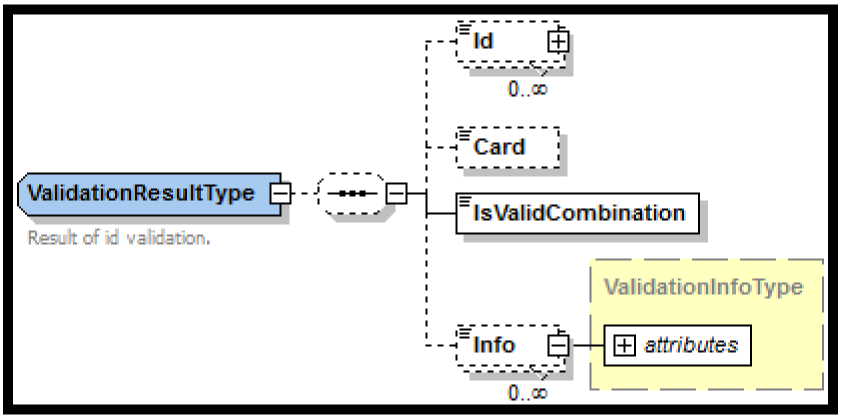


This type is used to handle extra information on the encountered business errors returned by the target service (case of unsuccessful completion of the request). The element **Value** of the **ProviderInfoType** has two possible values:

- 'NO_DATA_FOUND' for a reason specified by the Description and Information elements, the service has not been rendered.
- 'NO_RESULT', the service is denied due to a validation error or unfulfilled requirements.

Non-exhaustive list of the error messages of the target service is provided in the section 8 of the document and the examples show how the responses are transformed depending on the result of the verification.

- **ValidationResultType**



When the status code of the request status is ‘*Success’*, the element ValidationResult is returned, accessible and usable by the client application.

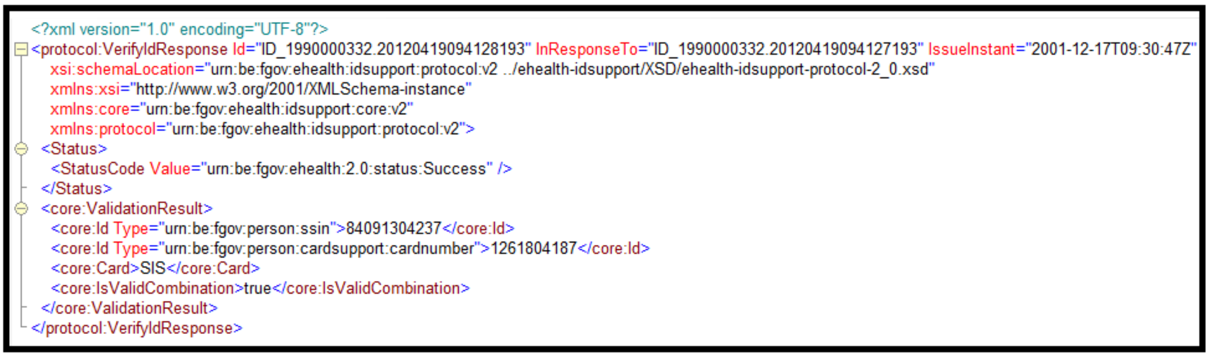
If the status of the card is correct and found in DB, the value of the element **isValidCombination** is set to ‘**true**’ and **card** element specifies the type of card.
Otherwise (invalid combination), **isValidCombination** is set to ‘**false**’ and the info element gives the reason.

Element	Attributes		Comments
	Id	Identifier of the response within the responder system.	Mandatory. The response message must contain a unique Id for tracing. The service generates this Id upon arrival of the request and use this for all generated log records linked to that request.
	InResponseTo	Id attribute of the request	Optional, if the request does not contain an Id, *InResponseTo* is left empty.
	IssueInstant	Date and time of the request.	Mandatory. Format YYYY-MM-DDThh:mm:ssZ
Status [1]		Provide the information on business errors.	The status element contains a recursive status code (mandatory). Optionally, the corresponding message describing the status and the detail (free type) available for services to put any element in it to give extra information on the encountered business errors returned by the target service (provider info)[8].
ValidationResult [0-1]		Returned when the level 1 Status code of the response is *'urn:be:fgov:ehealth:2.0:status:Success'*	Given the verification information, the status and the information of the combination of INSS and card support (**isValidation** is 'true' or 'false'), the type of the card support.

[8] *Please refer to the paragraph 5.2.1.2 and 5.2.2.3.1*

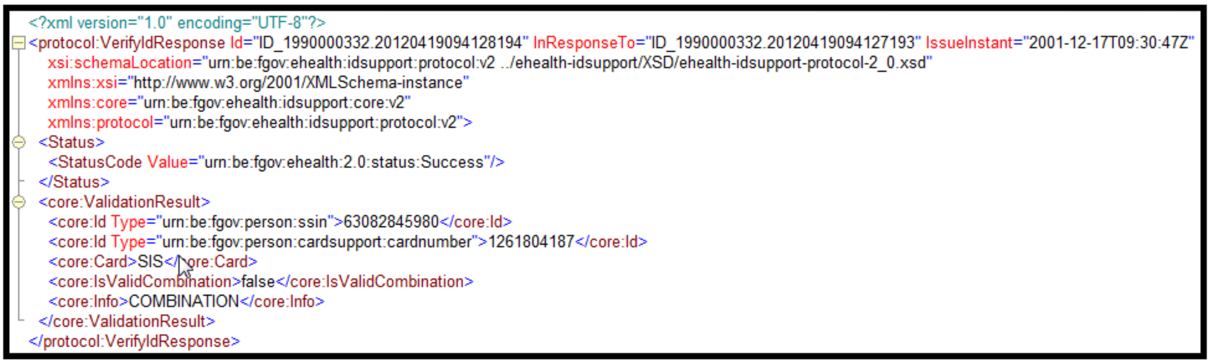
5.2.2.3.2 Examples of VerifyIdResponse.xml

- **Successful completion with valid combination INSS** v/s **card number**



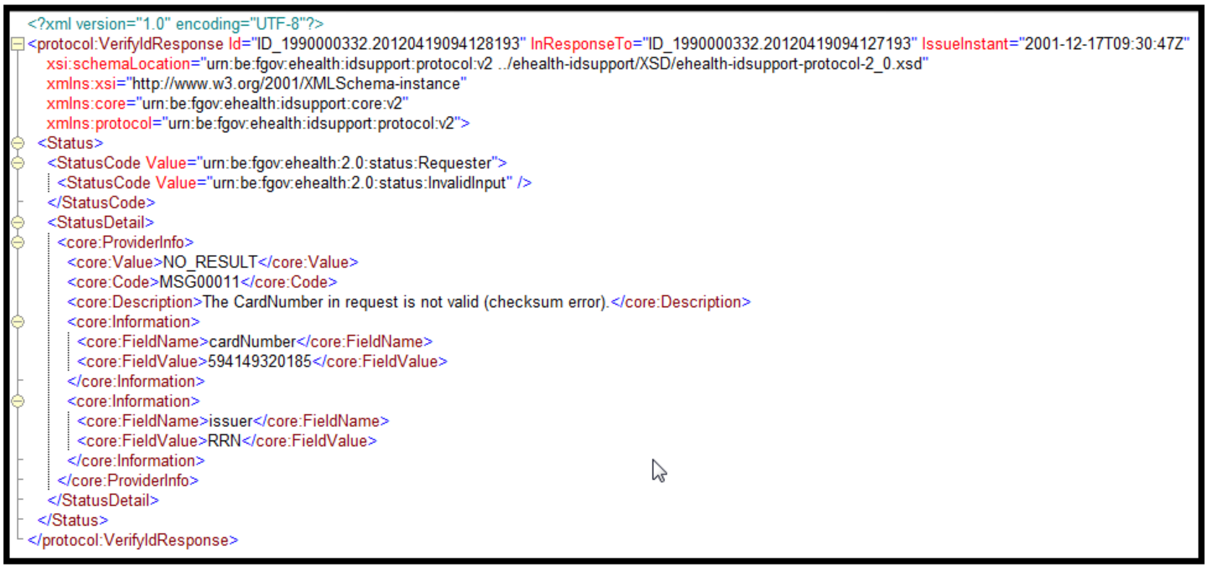
```
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdResponse Id="ID_1990000332.20120419094128193" InResponseTo="ID_1990000332.20120419094127193" IssueInstant="2001-12-17T09:30:47Z"
  xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ../ehealth-idsupport/XSD/ehealth-idsupport-protocol-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
  xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
 <Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Success" />
 </Status>
 <core:ValidationResult>
  <core:Id Type="urn:be:fgov:person:ssin">84091304237</core:Id>
  <core:Id Type="urn:be:fgov:person:cardsupport:cardnumber">1261804187</core:Id>
  <core:Card>SIS</core:Card>
  <core:IsValidCombination>true</core:IsValidCombination>
 </core:ValidationResult>
</protocol:VerifyIdResponse>
```

- **Successful completion with invalid combination INSS** v/s **card number.**



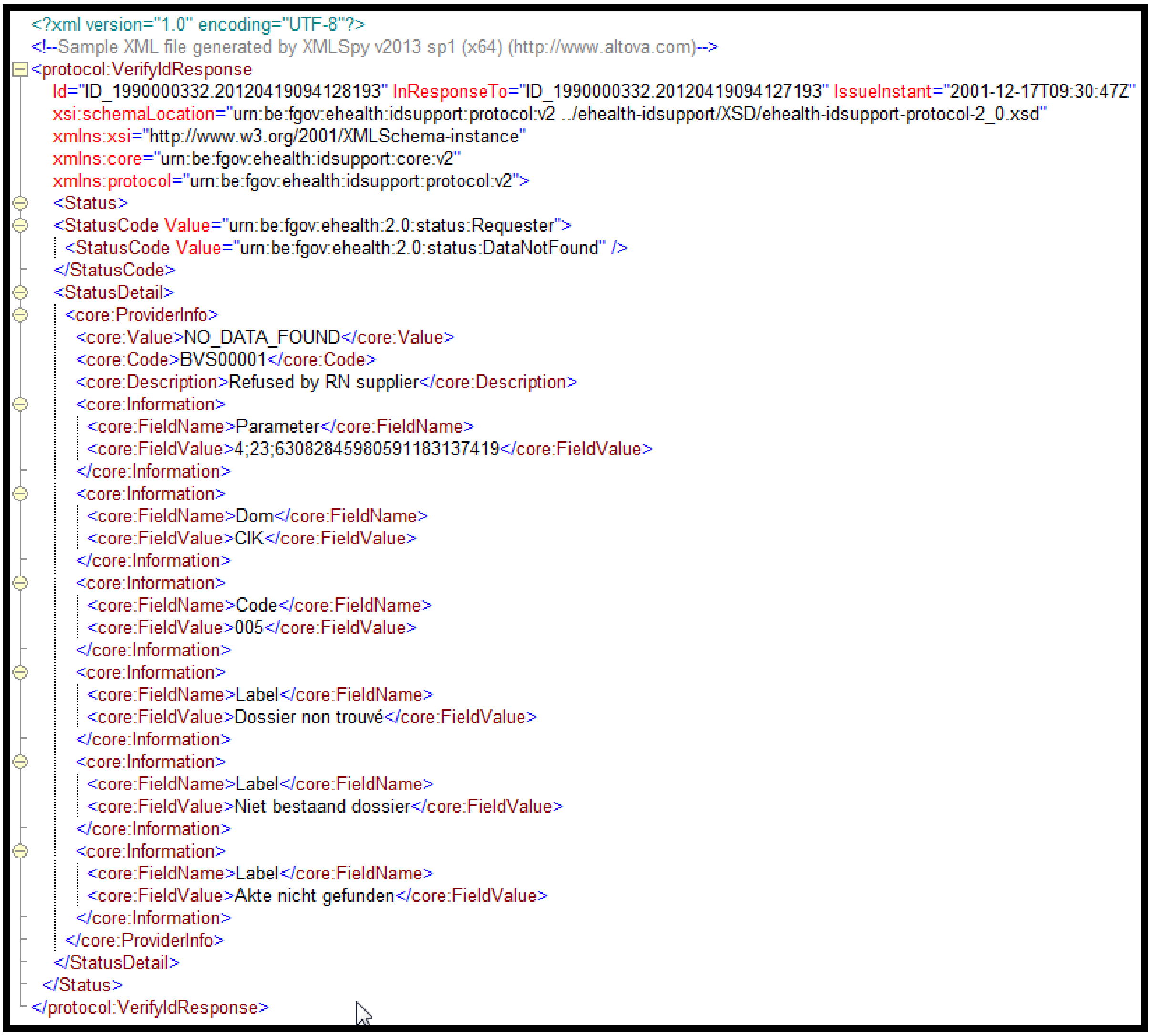
```
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdResponse Id="ID_1990000332.20120419094128194" InResponseTo="ID_1990000332.20120419094127193" IssueInstant="2001-12-17T09:30:47Z"
  xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ../ehealth-idsupport/XSD/ehealth-idsupport-protocol-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
  xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
 <Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Success"/>
 </Status>
 <core:ValidationResult>
  <core:Id Type="urn:be:fgov:person:ssin">63082845980</core:Id>
  <core:Id Type="urn:be:fgov:person:cardsupport:cardnumber">1261804187</core:Id>
  <core:Card>SIS</core:Card>
  <core:IsValidCombination>false</core:IsValidCombination>
  <core:Info>COMBINATION</core:Info>
 </core:ValidationResult>
</protocol:VerifyIdResponse>
```

- **Unsuccessful completion - NO_RESULT**



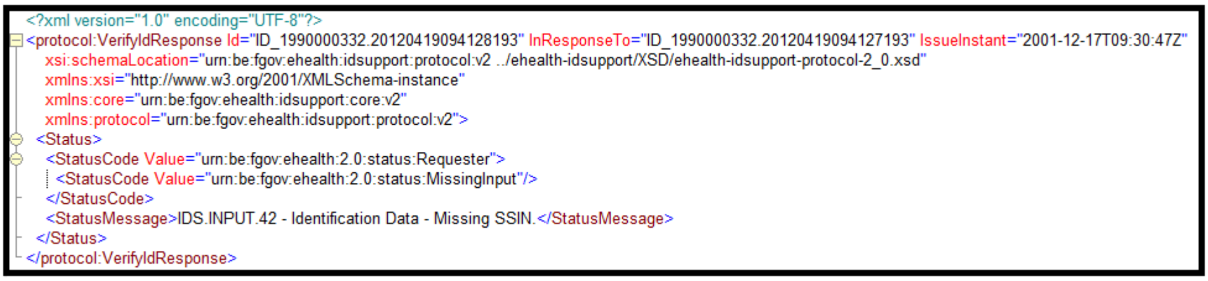
```
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdResponse Id="ID_1990000332.20120419094128193" InResponseTo="ID_1990000332.20120419094127193" IssueInstant="2001-12-17T09:30:47Z"
  xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ../ehealth-idsupport/XSD/ehealth-idsupport-protocol-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
  xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
 <Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Requester">
   <StatusCode Value="urn:be:fgov:ehealth:2.0:status:InvalidInput" />
  </StatusCode>
  <StatusDetail>
   <core:ProviderInfo>
    <core:Value>NO_RESULT</core:Value>
    <core:Code>MSG00011</core:Code>
    <core:Description>The CardNumber in request is not valid (checksum error).</core:Description>
    <core:Information>
     <core:FieldName>cardNumber</core:FieldName>
     <core:FieldValue>594149320185</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>issuer</core:FieldName>
     <core:FieldValue>RRN</core:FieldValue>
    </core:Information>
   </core:ProviderInfo>
  </StatusDetail>
 </Status>
</protocol:VerifyIdResponse>
```

- **Unsuccessful completion - NO_DATA_FOUND**



```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2013 sp1 (x64) (http://www.altova.com)-->
<protocol:VerifyIdResponse
  Id="ID_1990000332.20120419094128193" InResponseTo="ID_1990000332.20120419094127193" IssueInstant="2001-12-17T09:30:47Z"
  xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ../ehealth-idsupport/XSD/ehealth-idsupport-protocol-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
  xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
  <Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Requester">
   <StatusCode Value="urn:be:fgov:ehealth:2.0:status:DataNotFound" />
  </StatusCode>
  <StatusDetail>
   <core:ProviderInfo>
    <core:Value>NO_DATA_FOUND</core:Value>
    <core:Code>BVS00001</core:Code>
    <core:Description>Refused by RN supplier</core:Description>
    <core:Information>
     <core:FieldName>Parameter</core:FieldName>
     <core:FieldValue>4;23;63082845980591183137419</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>Dom</core:FieldName>
     <core:FieldValue>CIK</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>Code</core:FieldName>
     <core:FieldValue>005</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>Label</core:FieldName>
     <core:FieldValue>Dossier non trouvé</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>Label</core:FieldName>
     <core:FieldValue>Niet bestaand dossier</core:FieldValue>
    </core:Information>
    <core:Information>
     <core:FieldName>Label</core:FieldName>
     <core:FieldValue>Akte nicht gefunden</core:FieldValue>
    </core:Information>
   </core:ProviderInfo>
  </StatusDetail>
 </Status>
</protocol:VerifyIdResponse>
```

- **Unsuccessful completion – Missing mandatory information**



```
<?xml version="1.0" encoding="UTF-8"?>
<protocol:VerifyIdResponse Id="ID_1990000332.20120419094128193" InResponseTo="ID_1990000332.20120419094127193" IssueInstant="2001-12-17T09:30:47Z"
  xsi:schemaLocation="urn:be:fgov:ehealth:idsupport:protocol:v2 ../ehealth-idsupport/XSD/ehealth-idsupport-protocol-2_0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:core="urn:be:fgov:ehealth:idsupport:core:v2"
  xmlns:protocol="urn:be:fgov:ehealth:idsupport:protocol:v2">
 <Status>
  <StatusCode Value="urn:be:fgov:ehealth:2.0:status:Requester">
   <StatusCode Value="urn:be:fgov:ehealth:2.0:status:MissingInput"/>
  </StatusCode>
  <StatusMessage>IDS.INPUT.42 - Identification Data - Missing SSIN.</StatusMessage>
 </Status>
</protocol:VerifyIdResponse>
```

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 web service, the partner may obtain support from the contact center. (See chapter 3)

In case the eHealth platform 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 and body. This will allow eHealth to verify the integrity of the message and the identity of the message author.
- No encryption on the message.

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 acceptation or production.

7.1.1 Initiation

If you intend to use the eHealth platform 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 in the technical library on the eHealth platform portal.

The eHealth platform recommends performing tests of the WS in Acceptance environment first.

7.1.3 Release procedure

When development tests are successful, you can request to access the eHealth platform 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 platform 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 platform service for one of his 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

- ***VerifyId***: successful verification of person INSS and his card support number. In addition, the organization should also run negative test cases.

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, it is contained in a business response that undergoes a regular transformation[9] (see table 1, 2, 3, 4, 5 of the chapter 8.1 '*Business errors*').
- In the case of a technical error, a SOAP fault exception is returned (see table 6).

8.1 Business errors

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

Table 1: Description of common Id Support V2 business errors

Error type	Code	Description
IDS2.ACCESS Permission	IDS2.ACCESS.15	Invalid Legal context – Missing.
	IDS2.ACCESS.16	Invalid Legal context – Invalid.
IDS2.INPUT Invalid Input	IDS2.INPUT.37	Identification data - Invalid type combination (expecting either [inss and card number] or [barcoded]).
	IDS2.INPUT.52	Identification data - Invalid type.
	IDS2.INPUT.53	Identification data - Format error.
	IDS2.INPUT.54	Identification data - Checksum error.

Table 2: Result of successful completion **– Card SIS**[10]

IsValidCombination	***Info***
False	COMBINATION
False	SISCARDREVOKED1
False	SISCARDREVOKED2
False	VALIDITY
False	OTHER SIS CARD DELIVERED
True	

[9] *Please refer to the paragraph 5.2.2.3.1*

[10] *Colum names are corresponding to element names described in the paragraph 5.2.2.3.1* **ValidationResultType**

Table 3: Result of successful completion with invalid combination **- Card ISI+**[14]

IsValidCombination	***Info***
False	COMBINATION
False	lost
False	stolen
False	destroyed
False	expired
False	dead
False	old card
False	cancelled
False	card with invalid status
False	data identification has changed
False	undefined
True	

Table 4: Result of successful completion with invalid combination **– Card e-ID, Kidscard, foreign card, residence permit**[11]

IsvalidCombination	***Info***
False	COMBINATION
False	lost
False	stolen
False	destroyed
False	expired
False	dead
False	old card
False	cancelled
False	card with invalid status
False	undefined
True	

[11] *Colum names are corresponding to element names described in the paragraph 5.2.2.3.1* **ValidationResultType**

Table 5: Result of unsuccessful completion[12]

Value	***Code***	***Description***
NO_DATA_FOUND	MSG00000	Validation error
	BVS00001	Refused by RN supplier
NO_RESULT	IDS00011	The CardNumber in request is not valid (checksum error).

8.2 WS-I Basic Profile 1.1 - Errors

Your request must be WS-I compliant (See External Ref). If not you will receive one of the errors SOA-03001 – SOA-03003.

SOA-03001	***Malformed message***	*Consumer*	*This is the default error for content related errors in case no more details are known.*
SOA-03002	***Message must be SOAP***	*Consumer*	*Message does not respect the SOAP standard.*
SOA-03003	***Message must contain SOAP body***	*Consumer*	*Message respects the SOAP standard, but body is missing.*

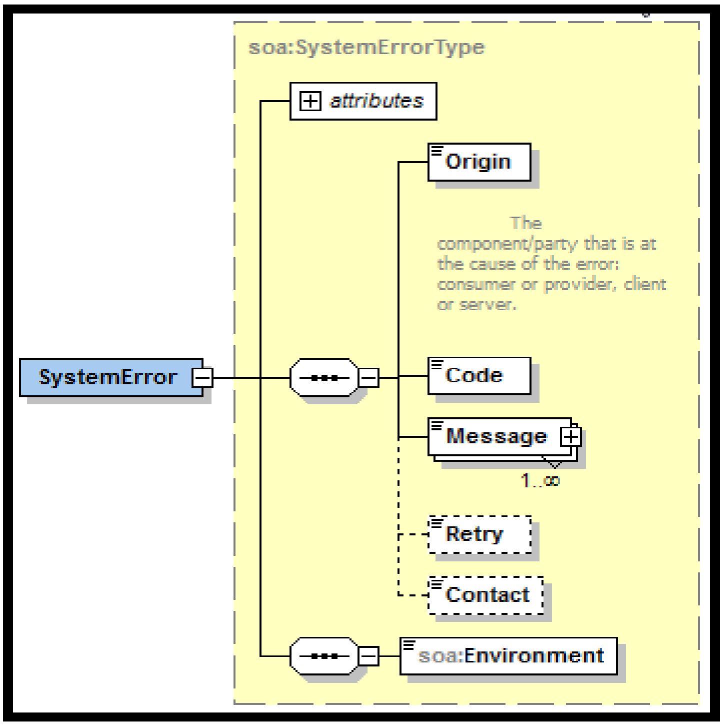
[12] *Colum names are corresponding to element names described in the paragraph 5.2.2.3.1* **ProviderInfoType**

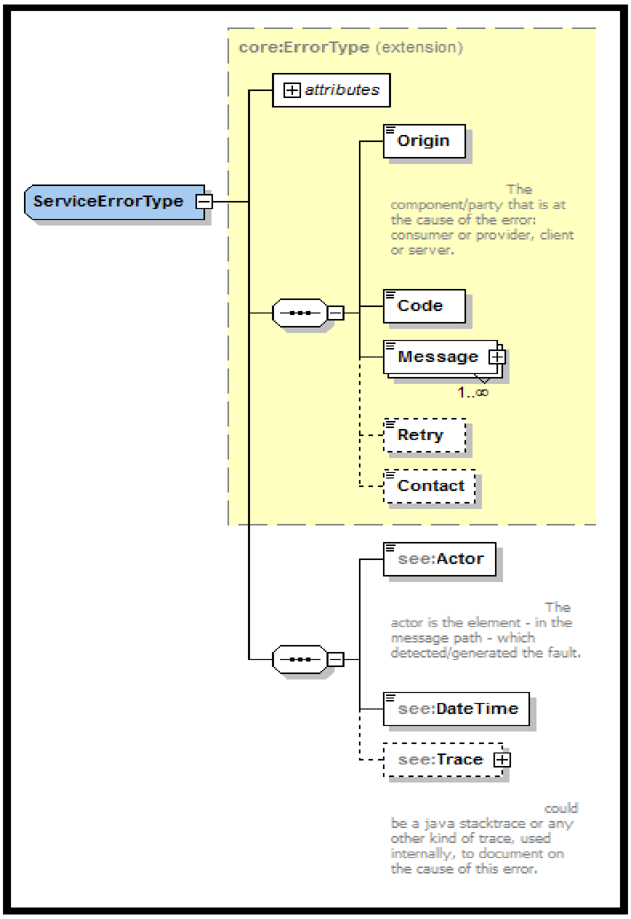
8.3 Technical errors

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 System- and BusinessErrors, thrown as SOAP Faults.

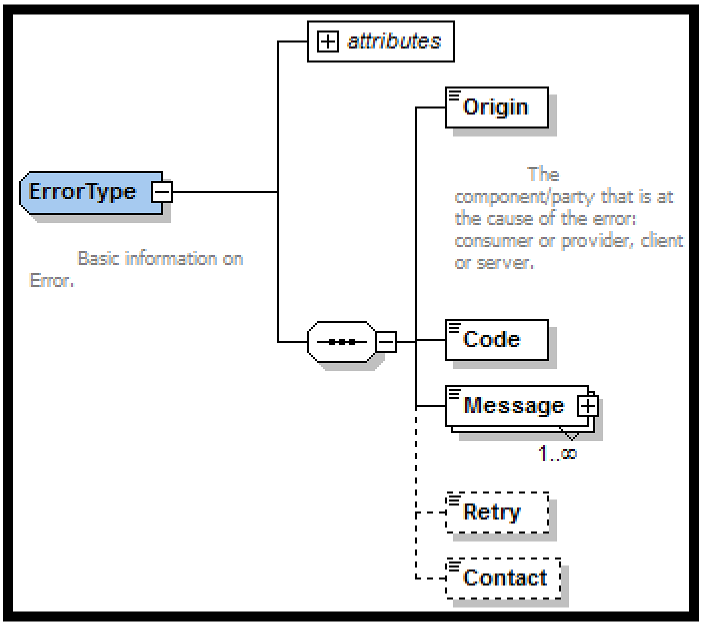
A **SystemError** MUST be thrown when a system failure occurred. It is not related to the business of the service. The SOA system error structure is as follows:

- **ehealth-errors-chema-soa-1_1.xsd**


soa:SystemErrorType
attributes
Origin
The component/party that is at the cause of the error: consumer or provider, client or server.
SystemError
Code
Message
1..∞
Retry
Contact
soa:Environment


core:ErrorType (extension)
attributes
ServiceErrorType
Origin
The component/party that is at the cause of the error: consumer or provider, client or server.
Code
Message
1..∞
Retry
Contact
see:Actor
The actor is the element - in the message path - which detected/generated the fault.
see:DateTime
see:Trace
could be a java stacktrace or any other kind of trace, used internally, to document on the cause of this error.

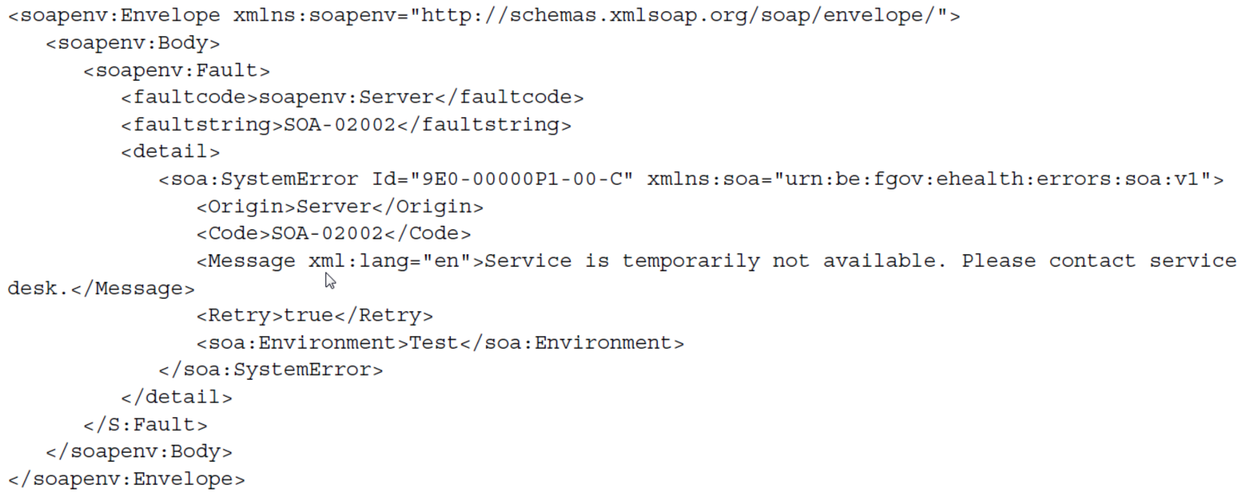
- **ehealth-errors-schema-core-1_1.xsd**


attributes
ErrorType
Basic information on Error.
Origin
The component/party that is at the cause of the error: consumer or provider, client or server.
Code
Message
1..∞
Retry
Contact

The SystemError element MUST contain a unique Id attribute for tracing. If the SystemError is a result of a SystemError SOAP Fault thrown by the Target Service, the Id must be the same as in that SystemError (See section 'Target Service Errors'.
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:ehealth: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	Optionality
faultcode	A code for identifying the fault	Mandatory
faultstring	A human readable explanation of the fault	Mandatory
faultactor	Information about who caused the fault to happen (the origin)	Optional
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.	Optional

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	The message was incorrectly formed or contained incorrect information.
server	There was a problem with the server so the message could not proceed.

Table 6: 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 no more details are known.
SOA-01001	Consumer	Service call not authenticated	From the security information provided • 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 • Retries will not work • Service desk may help with root cause analysis
SOA-02002	Provider	Service temporarily not available. Please try later	• 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): • Extended checks on the element format failed • Cross-checks between fields failed

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

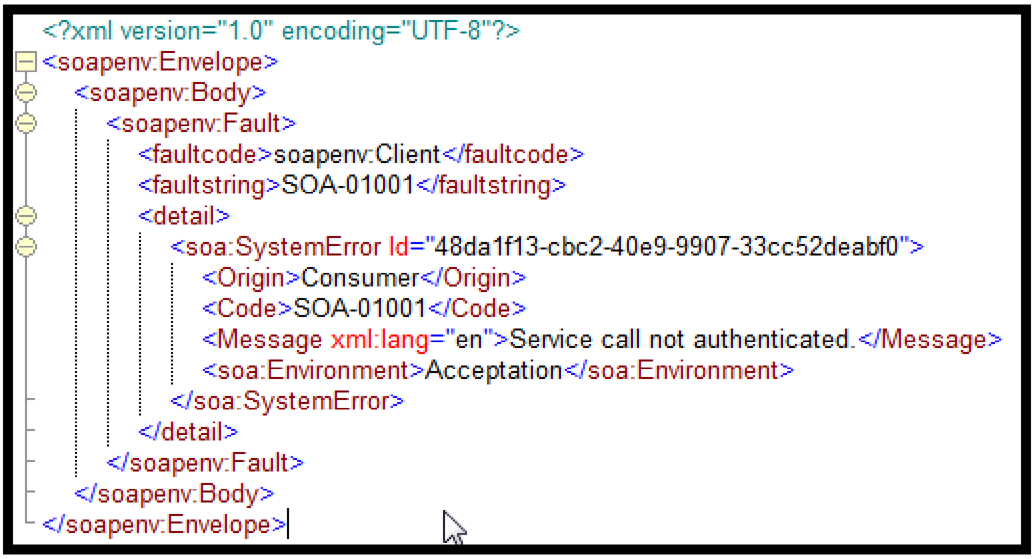
<soapenv:Header>

<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. It should be provided (when available) when requesting support.

Example:



```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope>
    <soapenv:Body>
        <soapenv:Fault>
            <faultcode>soapenv:Client</faultcode>
            <faultstring>SOA-01001</faultstring>
            <detail>
                <soa:SystemError Id="48da1f13-cbc2-40e9-9907-33cc52deabf0">
                    <Origin>Consumer</Origin>
                    <Code>SOA-01001</Code>
                    <Message xml:lang="en">Service call not authenticated.</Message>
                    <soa:Environment>Acceptation</soa:Environment>
                </soa:SystemError>
            </detail>
        </soapenv:Fault>
    </soapenv:Body>
</soapenv:Envelope>
```