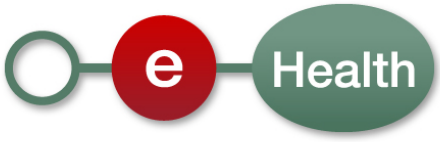


Cookbook
UREG
(Urgentie registratie – Enregistrement urgence)
Version 1.3

eHealth platform

Willebroekkaai 38,
Quai de Willebroek

1000 BRUSSELS

Table of contents

1 Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
1	18/03/2013	eHealth	First version
2	23/05/2014	eHealth	Second version
3	28/11/2014	eHealth	Third version
4	25/02/2015	eHealth	Fourth version

2 Introduction

2.1 Goal of the service

The goal of the UREG project is to collect in real time information about emergency services, their patients, their pathologies, their activity, … and allowing punctual survey on a short time (and also sometimes on a limited area) about punctual subjects (epidemics, number of occurrences of a pathology on a given amount of time, ….).

The UREG Project will focus on the following points:

- Activity of emergency services
- Quality in emergency services
- Sanitary control
- Feed-back
- Scheduling and Financing
- Analysis initiated by the Secretary of the Public Health or by scientists instances of emergency care

The encoding of the data of the emergency services must not create an overload of work. Most of the data listed here are still encoded in the hospital systems. The FPS will have to determinate the structure and the format of the required data.

The hospital will send those data in the required format through web services; in two possible ways (optionally a third way can be developed by the hospitals):

- Integration of a WS client in the backend of the hospital
- Generation of XML files collected by a file grabber using the WS client
- Optionally, a full application can be developed by the hospitals

2.2 Goal of the document

This document provides functional and technical information and allows an organization to integrate and use the eHealth service.

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

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

2.3 eHealth document references

All the document references can be found in the technical library on the eHealth portal[1]. These versions or any following versions can be used for the eHealth service.

ID	Title	Link	Version	Date	Author
1	Glossary.pdf	/	1.0	01/01/2010	eHealth

2.4 External document references

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

ID	Title	Source	Date	Author
1	UREG FPS	***http://www.health.belgium.be/eportal/Healthcare/Healthcarefacilities/Registrationsystems/UREG/Guidelines/***	N.A	Federal Public Services (FPS)

[1] ***www.ehealth.fgov.be***

3 Support

3.1 Certificates

An eHealth certificate is used to identify the initiator of the request.

The acceptance certificate to be used in UREG may only be requested by hospitals. Each hospital willing to perform acceptance tests must have a certificate-responsible who needs to be registered at the FOD Gezondheid/SPF Santé. Please contact therefore the below mentioned business contact point (section 3.2 of this document).

Only after confirmed registration at the FOD/SPF, the hospital is entitled to proceed with the acceptance certificate request. Please refer to the eHealth portal to launch the certificate request in the appropriate (Acceptance resp. Production) environment.

Dutch version:
https://www.ehealth.fgov.be/nl/support/basisdiensten/ehealth-certificaten

French version:
https://www.ehealth.fgov.be/fr/support/services-de-base/certificats-ehealth

For technical issues regarding eHealth certificates

Acceptance: ***acceptance-certificates@ehealth.fgov.be***

Production: ***support@ehealth.fgov.be***

3.2 Contact

For all **business** questions, please contact FOD Gezondheid/SPF Santé :

- Dutch version:
  info.ureg@gezondheid.belgie.be
- French version:
  info.ureg@sante.belgique.be
- Phone: 02/524 85 14

For technical issues in production

eHealth ContactCenter:

- Phone: 02/788 51 55
- Mail: ***support@ehealth.fgov.be***
- *Contact Form :*

https://www.ehealth.fgov.be/nl/neem-contact-met-de-openbare-instelling-eHealth-platform (Dutch)

https://www.ehealth.fgov.be/fr/contactez-institution-publique-plate-forme-eHealth (French)

FOR PARTNERS AND SOFTWARE DEVELOPERS ONLY

- For technical issues in production please contact: ***support@ehealth.fgov.be or call 02/788 51 55***
- For technical issues in acceptance please contact: ***Integration-support@ehealth.fgov.be***

4 Technical requirements

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

Dutch version: https://www.ehealth.fgov.be/nl/support/connectors
French version: https://www.ehealth.fgov.be/fr/support/connectors

Alternatively, you can write your own implementation. The usage of the Secure Token Service (STS) and the structure of the exchanged XML-messages are described in the eHealth STS cookbook.

Dutch version: https://www.ehealth.fgov.be/nl/support/sts-secure-token-service
French version: https://www.ehealth.fgov.be/fr/support/sts-secure-token-service

4.1 Use of the eHealth SSO solution

The complete overview of the profile and a step-by-step description of how to protect a new application with the SSO @ eHealth are described in the eHealth SSO cookbook.

Dutch version: ***https://www.ehealth.fgov.be/nl/support/basisdiensten/toegangsbeheer/identity-authorization-management-iam***
French version: ***https://www.ehealth.fgov.be/fr/support/services-de-base/gestion-des-acces/identity-authorization-managment-iam***

This section specifies how to obtain a SAML token from the STS (Secure Token Service) in order to have access to the UREG web service. You must specify several attributes in the request:

For a hospital, the needed identification attributes are the following:

- **The NIHII number as identifier of the hospital**
  (namespace: “urn:be:fgov:identification-namespace”)
  “urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number” and
  “urn:be:fgov:ehealth:1.0:hospital:nihii-number”
- **The identification number of the campus**
  (namespace: “urn:be:fgov:identification-namespace”)
  “urn:be:fgov:ehealth:1.0:campus:site-number”

For a hub, the needed identification attributes are the following:

- **The EHP number as identifier of the hub**
  (namespace: “urn:be:fgov:identification-namespace”)
  “urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number” and
  “urn:be:fgov:ehealth:1.0:hub:ehp-number”

You have also to precise which information must be validated by eHealth.

For a hospital, the needed attributes are the following:

- **The NIHII number as identifier of the hospital**
  (namespace: “urn:be:fgov:identification-namespace”)
  “urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number” and
  “urn:be:fgov:ehealth:1.0:hospital:nihii-number”

- **The identification number of the campus**
  (namespace: “urn:be:fgov:identification-namespace”)
  "urn:be:fgov:ehealth:1.0:campus:site-number"
- **Attribute to verify if it is a recognized hospital**
  (namespace: "urn:be:fgov:certified-namespace:ehealth")
  “urn:be:fgov:ehealth:1.0:certificateholder:hospital:nihii-number:recognisedhospital:boolean”
- **Attribute to verify if it is a recognized site**
  (namespace: "urn:be:fgov:certified-namespace:ehealth")
  “urn:be:fgov:ehealth:1.0:campus:site-number:recognisedsite:boolean”

For a hub, the needed attributes are the following:

- **The EHP number as identifier of the hub**
  (namespace: “urn:be:fgov:identification-namespace”)
  “urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number” and
  “urn:be:fgov:ehealth:1.0:hub:ehp-number”
- **Attribute to verify if it is a recognized hub**
  (namespace: "urn:be:fgov:certified-namespace:ehealth")
  “urn:be:fgov:ehealth:1.0:certificateholder:organization:ehp-number:recognisedhub:boolean”

4.1.1 Encryption

4.1.1.1 The encryption to be used is based on the End-To-End Encryption library provided by eHealth

- Information regarding the End-To-End encryption service can be found at the following address:
  - Dutch version

    https://www.ehealth.fgov.be/nl/support/basisdiensten/systeem-voor-end-end-vercijfering
  - French version

    https://www.ehealth.fgov.be/fr/support/services-de-base/systeme-de-cryptage-end-to-end

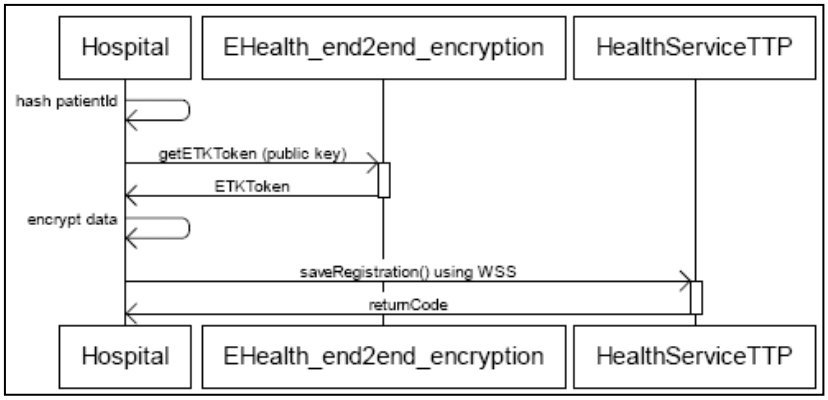
  At these addresses, you can download example of java code for making encryption a message
- The end-to-end encryption process used uses two keys: the private key of the hospital and the public key of the FPS. The private key of the hospital is used to seal the message; the public key of the FPS is used to actually encrypt the message.
- The public key of the FPS can be retrieved from the ETK depot, a service offered by eHealth
- Use the following input parameters to retrieve the public key in ETK-depot:
  - Type: CBE
  - Value: 0367303762
  - ApplicationId: UREG

4.1.2 Security policies to apply

For security reasons and respect of the patient’s privacy, all messages must follow these rules:

- In case of a questionnaire answer of type “patient”:
  - The patients Id has to be hashed by the hospital (variable named “patientId”)
  - The hospitals are free to choose whatever hashing they prefer.
- For every questionnaire answer that is being transmitted to the UREG service, a unique registration Id has to be assigned (variable named “registrationId”)
  - The registrationId has to be hashed by the hospital
  - The hospital can choose whatever hashing they prefer
  - Whenever a new answer to the same questionnaire for this same registration is transmitted, the same original registrationId is to be used

  An example of such a scenario would be when the hospital sends the data a first time upon patient’s entry and sends the (more complete) data a second time upon patients discharge.
- Use of SSL : calls to E-Health – SSL 1 way
- Use of WSS :
  - The call to the eHealth web service is protected by WSS. This means that the sent message is signed by the private key of the hospital.
  - The signature is required on the following parts of the message:
    - Body
    - Timestamp
    - Token



4.2 XSD, WSDL

The XSD and WSDL of the UREG service can be found in the service registry.

Please use the following link: ***https://services.ehealth.fgov.be/registry/uddi/bsc/web***

5 Overview

Different types of questionnaires can be defined by the business and assigned to the hospital sites. A questionnaire is a set of questions in XSD format.

A questionnaire can be related to:

- an emergency service;
- patients.

For both questionnaire types, there are:

- static questionnaires (which never change, or not very often);
- ad hoc questionnaires (which are temporary questionnaires targeted on specific topics for selected hospital sites).

As each questionnaire defined at FPS level (independently of its type) is intended to be delivered to one or more hospital sites for a defined interval of time, the hospital site needs to be able to:

- test the end-to-end connectivity (see 6.1)
- retrieve all questionnaires assigned to this hospital site (see 6.2)
- respond to those questionnaires (see 6.3)

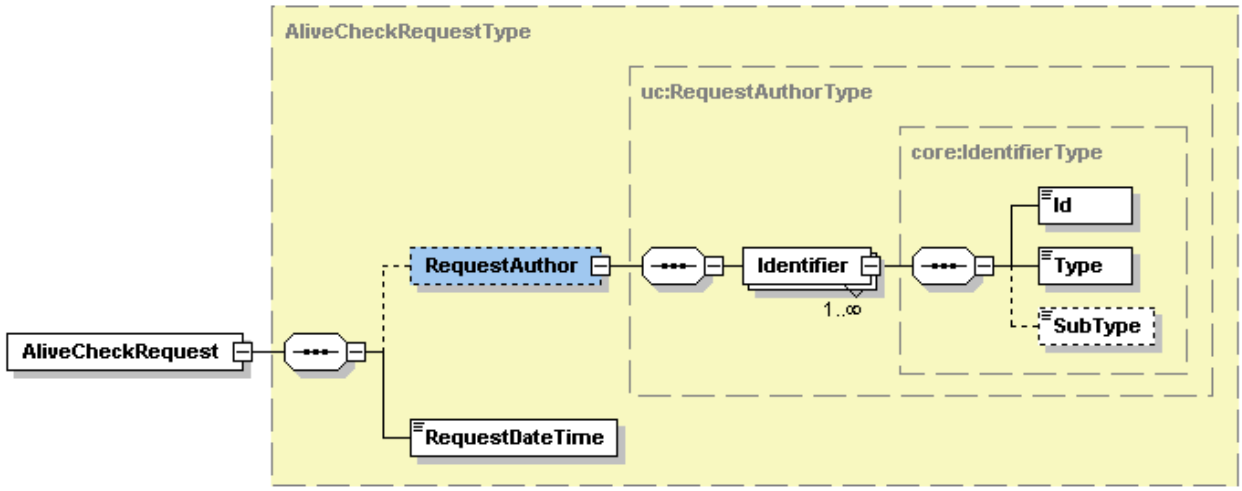
Those functionalities will be provided through web service in acceptation and production environment.

6 Web service

6.1 Method AliveCheck

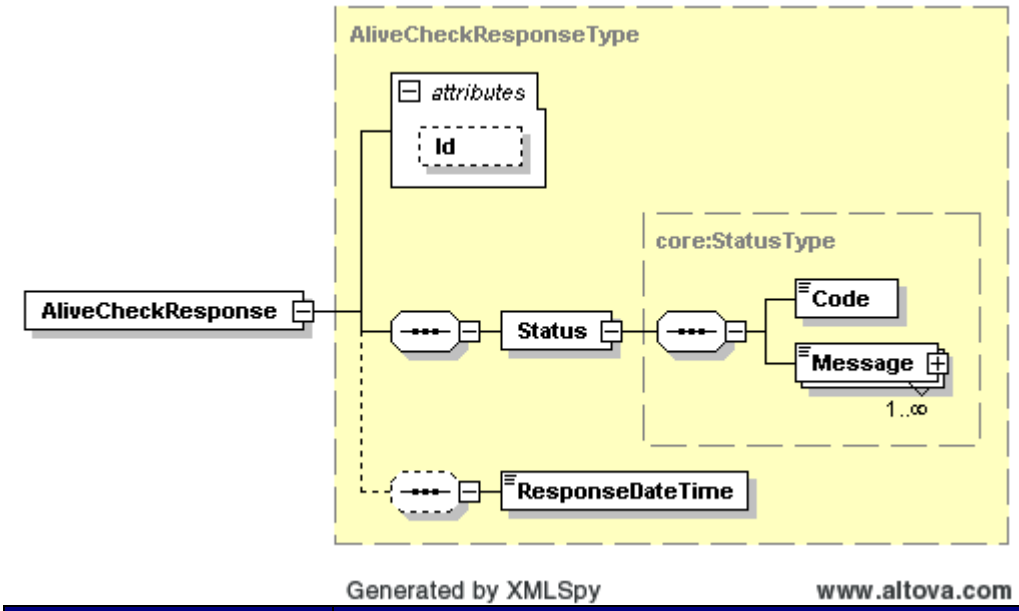
This method is a technical test, from hospital site to the FPS to check that the connection between hospital site and FPS is well established. If so, you will receive a reply from web service with a code value “0” and an empty message. Otherwise, you will receive a code <> 0 and a non-empty message with the error description.

6.1.1 Input arguments in AliveCheckRequest



Field name	Descriptions
RequestAuthor	Identity of the hospital and the site
RequestDateTime	Time and date of the request.

6.1.2 Output arguments in AliveCheckResponse



Field name	Descriptions
Id	The ticket number (*TicketNumber*) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	The *Status* block will contain a code and a message. If no error has occurred during the transaction, the *Code* will be '0' and the *Message* will be empty. Otherwise: • The *Code* will be an error code which unequivocally identifies the problem. A problem can be related to the infrastructure (availability of the web service ...) or content of the request. • The *Message* will be a description of the error
ResponseDateTime	Time and date of the response. Time and date format: YYYY-MM-DDThh:mm:ssZ

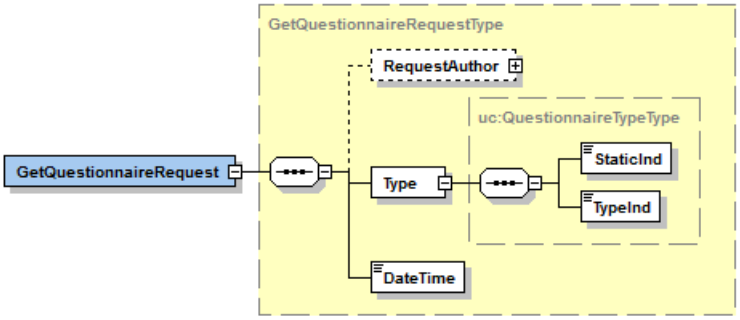
6.1.3 Example

Request	Response
<urn:AliveCheckRequest> <RequestAuthor> <Identifier> <Id>**77777766**</Id> <Type>**NIHII**</Type> <SubType>**HOSPITAL**</SubType> </Identifier> <Identifier> <Id>**1111**</Id> <Type>**SITE**</Type> </Identifier> </RequestAuthor> <RequestDateTime>2010-04-23T07:44:35Z</RequestDateTime> </urn:AliveCheckRequest>	<urn:AliveCheckResponse Id="bce0bb5f-83ff-4a56-a2db-2329a9624900" xmlns:urn="urn:be:fgov:ehealth:ureg:protocol:v1"> <Status> <Code>0</Code> <Message/> </Status> <ResponseDateTime>2014-11-28T09:56:21Z</ResponseDateTime> </urn:AliveCheckResponse>

6.2 Method GetQuestionnaire

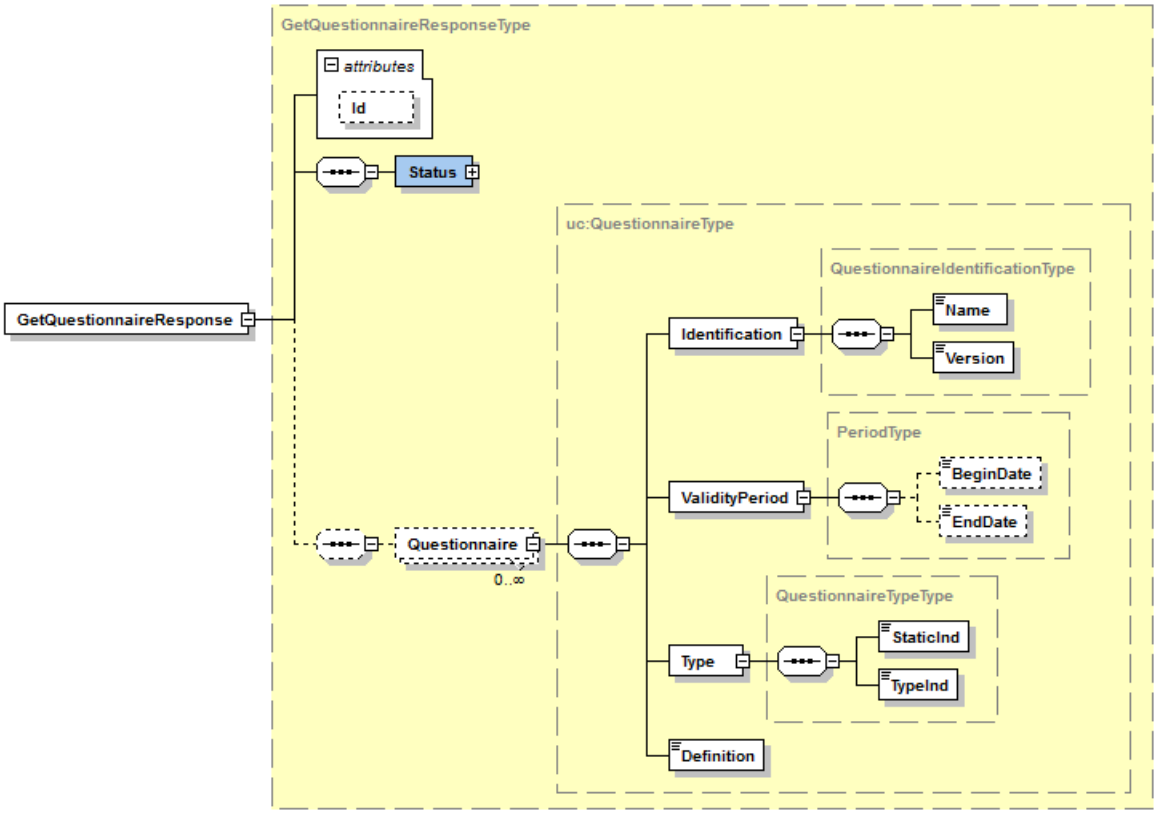
This method retrieves all questionnaires assigned to the calling hospital site, at a given date, for the specified types of questionnaires: static or ad Hoc and related to a patient or a service. All questionnaire definitions are in XSD format.

6.2.1 Input arguments in GetQuestionnaireRequest



Field name	Descriptions
RequestAuthor	Identity of the hospital and the site
Type/TypeInd	The type of questionnaire, uses the value 1 for patient type or the value 2 for service type or the value 0 for all
Type/StaticInd	Indicator for the static of ad hoc questionnaire type, uses the value 1 for static type or the value 2 for ad hoc type or the value 0 for all
DateTime	The time and date for which the questionnaire list should be valid, only valid questionnaires for this date will be returned.

6.2.2 Output arguments in GetQuestionnaireResponse



Field name	Descriptions
Id	The ticket number (*TicketNumber*) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	The *Status* block will contain a code and a message. If no error has occurred during the transaction, the *Code* will be '0' and the *Message* will be empty. Otherwise: • The *Code* will be an error code which unequivocally identifies the problem. A problem can be related to the infrastructure (availability of the web service, ...) or content of the request. • The *Message* will be a description of the error
Questionnaire	Zero or more questionnaires that are valid for the requested date. Each questionnaire will contain the following information: • name (Identification/Name) • Version (Identification/Version) • Validity-from date (ValidityPeriod/BeginDate) • Validity-end date (ValidityPeriod/EndDate) • Questionnaire for a patient or an emergency service (Type/TypeInd) • Questionnaire static or ad hoc (Type/StaticInd) • Definition: CDATA with the content in XSD format

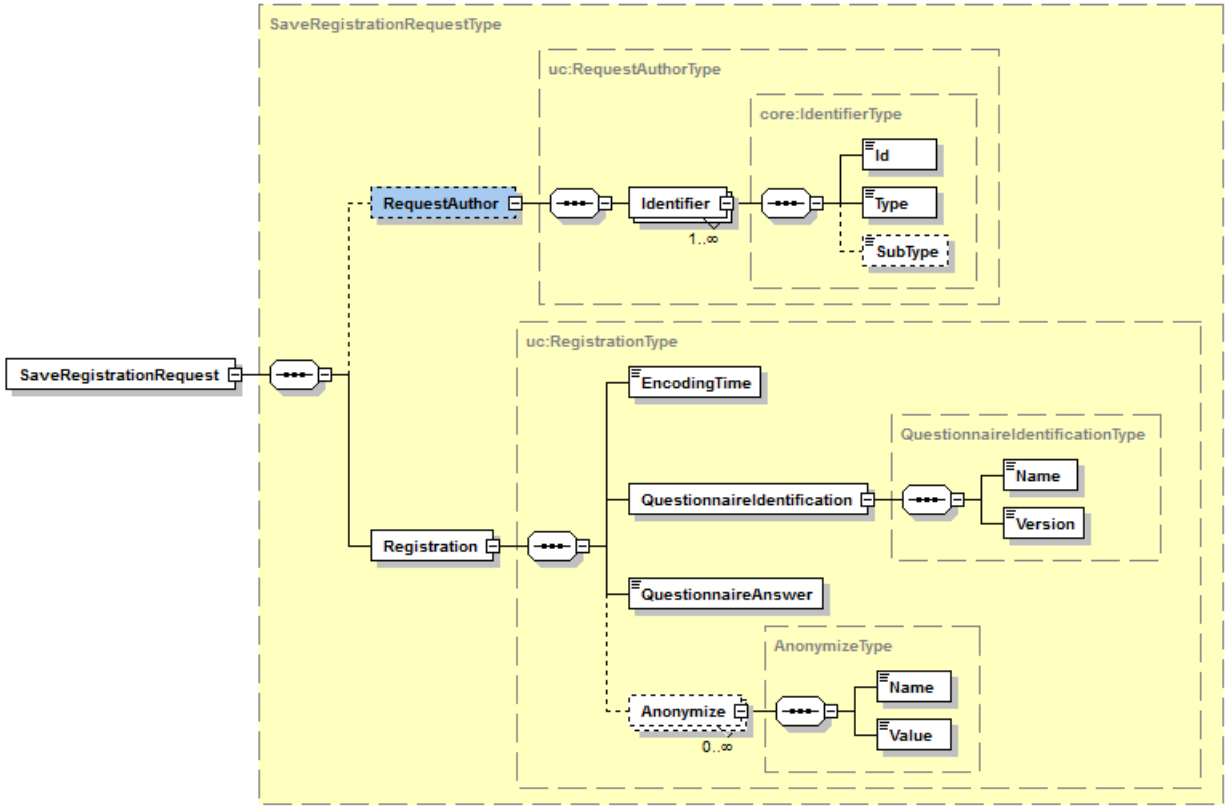
6.2.3 Example

Request	Response
<ureg:GetQuestionnaireRequest> <Type> <StaticInd>**1**</StaticInd> <TypeInd>**1**</TypeInd> </Type> <DateTime>**2009-12-15T12:12:00Z** </DateTime> </ureg:GetQuestionnaireRequest>	<urn:GetQuestionnaireResponse Id="6e768720-cf93-4b7a-ac71-61da3e2452d2" xmlns:urn="urn:be:fgov:ehealth:ureg:protocol:v1"> <Status> <Code>0</Code> <Message/> </Status> <Questionnaire> <Identification> <Name>ADMMED</Name> <Version>8</Version> </Identification> <ValidityPeriod> <BeginDate>2011-11-30T23:00:00Z</BeginDate> </ValidityPeriod> <Type> <StaticInd>1</StaticInd> <TypeInd>1</TypeInd> </Type> <Definition><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?><xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0">...]></Definition> </Questionnaire> </urn:GetQuestionnaireResponse>

6.3 Method SaveRegistration

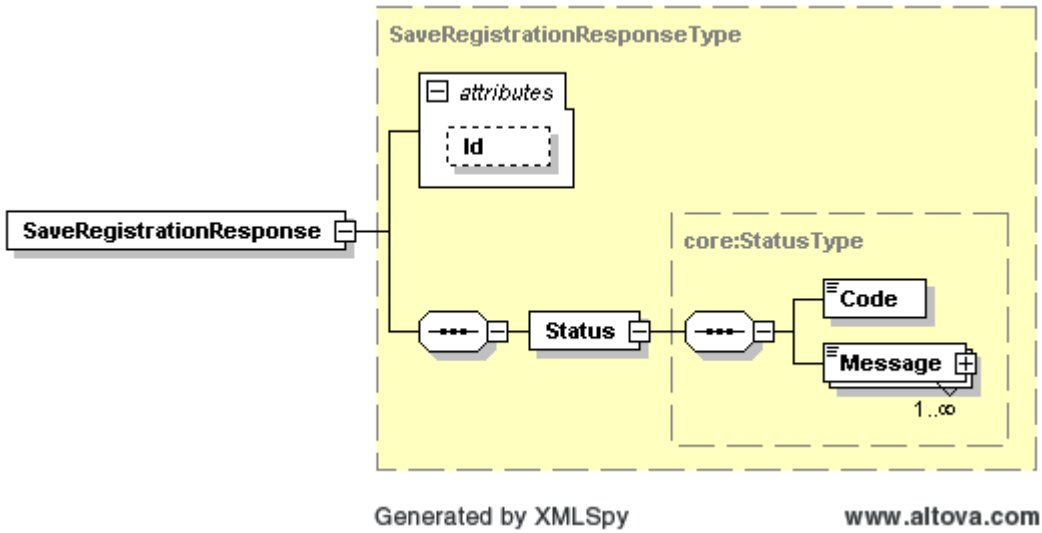
This method saves the registration for the selected questionnaire (registration has to be sent in XML format based on the received XSD). If the sent XML is well formed it will be accepted otherwise it will be rejected. This is the only synchronous validation. All validations about the contents of the XML will be processed asynchronously).

6.3.1 Input arguments in SaveRegistrationRequest



Field name	Descriptions
RequestAuthor	Identity of the hospital and the site
EncodingDateTime	The time and date the questionnaire is being answered for the first time. Time and date format: YYYY-MM-DDThh:mm:ssZ
Registration/QuestionnaireIdentification/Name	Questionnaire name
Registration/QuestionnaireIdentification/Version	Questionnaire version
Registration/QuestionnaireAnswer	The base64 encrypted answer to a questionnaire in XML format. For more information about encryption, see 4.1.1 The original questionnaires answer is an XML message that complies with the questionnaires definition (the questionnaires XSD). This (XML) answer is then encrypted and used here.
Registration/Anonymize/Name	Variables name of the variable that needs to be anonymized “patientId” for a patients Id and “registrationId” for the registration Id
Registration/Anonymize/Value	The actual value for the variable named above. Patient or registration identifier. The identifier must be hashed. Example : patient identifier is “test” => h(“test”) := “test_12” So the value to use is “test_12”

6.3.2 Output arguments in SaveRegistrationResponse



Field name	Descriptions
Id	The ticket number (*TicketNumber*) is attributed to the exchange request/response by the eHealth platform. This is used to identify the eHealth session.
Status	The *Status* block will contain a code and a message. If no error has occurred during the transaction, the *Code* will be '0' and the *Message* will be empty. Otherwise: • The *Code* will be an error code which unequivocally identifies the problem. A problem can be related to the infrastructure (availability of the web service, ...) or content of the request. • The *Message* will be a description of the error

6.3.3 Example

Request	Response
<urn:SaveRegistrationRequest> <RequestAuthor> <Identifier> <Id>**77777766**</Id> <Type>**NIHII**</Type> <SubType>**HOSPITAL**</SubType> </Identifier> <Identifier> <Id>**1111**</Id> <Type>**SITE**</Type> </Identifier> </RequestAuthor> <Registration> <EncodingTime>**2009-12-15T12:12:00Z**</EncodingTime> <QuestionnaireIdentification> <Name>**TEST**</Name> <Version>**1**</Version> </QuestionnaireIdentification> <QuestionnaireAnswer>**MIAGCSq...AAAA==**</QuestionnaireAnswer> <Anonymize> <Name>**patientId**<Name> <Value> **p_12**/Value> </Anonymize> <Anonymize> <Name>**registrationId**</Name> <Value>**r_12**</Value> </Anonymize> </Registration> </urn:SaveRegistrationRequest>	<urn:SaveRegistrationResponse Id="1a1e617b-69e3-4c9c-86d9-8a506062a6d3"> <Status> <Code>0</Code> <Message/> </Status> </urn:SaveRegistrationResponse>

7 Risks and security

7.1 Security

7.1.1 Business security

In case the development adds an additional use case based on an existing integration, eHealth 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 3.2).

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 web service 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.

7.1.2 Web service

Web service 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.

8 Test procedure

This chapter explains the procedure for testing UREG in acceptance environment.

8.1 Request a test case

To be authorized to call the web services, the hospital with its site or the hub must be configured in eHealth acceptance environment. So, create an excel file like the below example and send it to ***info@ehealth.fgov.be*** with the subject: "UREG request test case".

Example: hospital

Site identifier	**Site name**	**Hub identifier**	**Hub name**	**Hospital identifier**	**Hospital name**
1111	**Site test acc**	/	/	**99999971**	**Hospital test acc**

Example: hub

Site identifier	**Site name**	**Hub identifier**	**Hub name**	**Hospital identifier**	**Hospital name**
/	**/**	1111	Hub test acc	**/**	**/**

8.2 Request an hospital certificate

The developed functionality needs to be tested using an acceptance certificate for hospital. Therefore a participating test-hospital must firstly have a certificate-responsible who needs to be registered at the FOD Gezondheid/SPF Santé. Therefore please contact the above mentioned business contact point (see section 3.1). Only after confirmed registration at the FOD/SPF, this certificate-responsible is entitled to proceed with the acceptance certificate request.

Acceptance tests need to be performed on-site (in a pilot hospital). Therefore, the hospital-acceptance certificate is required.
The UREG project owner (FOD Gezondheid/SPF Santé) requires explicitly that the key pair and certificate in the acceptance-environment are managed and remain supervised by the certificate responsible of the hospital (i.e. delegation to an external person is prohibited).

Software companies may only conduct acceptance tests in the acceptance environment of the hospital where the acceptance certificate and key pair of the specific environment shall be consulted on the predefined path ("Home Directory" under:\ehealth\keystore\ as set out in eHealth Certificate Manager – manual § 2.1.12).

8.3 Call web services

When the test case and the certificate are created in the acceptance environment, you can send requests to the eHealth web services. We suggest performing the following test:

1. Request AliveCheckRequest (section 6.1)

2. Request GetQuestionnaireRequest (section 6.1.3). In the response, save the XSD contained in the questionnaire/definition and create an XML compliant with the XSD.
3. Request SaveRegistrationRequest (section 6.1.5.1). The XML created in step 2 must be encrypted with the FPS's UREG public key (section 4.1.1) and sealed with the hospitals private key before it is added in Registration/QuestionnaireAnswer of request.

9 Error and failure messages

There are different possible types of response:

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

If an error occurs, first please verify your request. Following table contains a list of common system error codes for the eHealth Service Bus.

Table 1: Description of the possible SOAP fault exceptions.

Error code	Component	Description	Solution/Explanation
SOA-00001	Unknown	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