TECHNICAL DOCUMENT

Web service technical documentation TESSy

Version 1.1

www.ecdc.europa.eu

ECDC TECHNICAL DOCUMENT

Web service technical documentation TESSy

Version 1.1

2009-08-07

TESSy Helpdesk contact details:

Phone support: +46 8 5860 1601

E-mail support: TESSy@ecdc.europa.eu

Table of contents

Version history

Version:	Change:
1.0	First version
1.1	Authentication details changed in Chapter 3.1 Authentication

Purpose

This document describes machine-to-machine interface to TESSy and how to use it. It describes what the TESSy web service is, what is required to use it, how to make requests and how to process the response. This document should be the main guideline when implementing the TESSy web service.

Audience

The indented reader of this document is anyone that wishes to implement and set up machine-to-machine communication with TESSy to automate the reporting of surveillance data to ECDC. The reader is required to have a basic knowledge of how web services works and should feel confident using terms like SOAP and XML.

1 Introduction

This document describes the TESSy web service and how to set up machine-to-machine communication with TESSy. It is basically divided into three parts:

- an introduction to the TESSy web service and machine-to-machine communication;
- a technical description of the method interface and how to understand responses from method calls; and
- a description of the TESSy test and TESSy production environments.

1.1 Background

The European Centre for Disease Prevention and Control (ECDC) was established in 2005. It is an EU agency that aims to strengthen Europe's defences against communicable disease. The founding document (Reg. No. 851/2004/EC) establishes that the ECDC shall provide a technical platform for data collection in Europe and cooperate with the competent bodies (institutions) recognised by Member States (MS), also in collection, analysis and dissemination of data.

Prior to 2005, there were 17 Dedicated Surveillance Networks (DSNs) that collected data on a variety of communicable disease. All EU MS submitted data individually to most DSNs, using different file specifications and reporting protocols. This demanded a large effort by MS. On the other hand, all DSNs had to provide systems for data collection, validation and analysis.

TESSy, The European Surveillance System, includes a database system for collection of data (also called TESSy), and groups of experts from ECDC and from the participating EU and non-EU countries (epidemiologists and data managers/IT experts, microbiologists and statisticians), hundreds of professionals who report diseases, laboratories, and other expert profiles involved in the surveillance of communicable diseases at the European level. This network of experts has been formed based on the ECDC founding regulation in bilateral working relationships between ECDC, each MS and other Community and partner institutions. This process has also established the procedure for identifying/requesting user access to the system. More information on this can be obtained from TESSy Helpdesk.

TESSy is a highly flexible metadata-driven system for collection, validation, cleaning, analysis and dissemination of data on diseases under EU-wide surveillance. The key aims of the system are data analysis and information outputs for public health action. All EU Member States (27) and EEA countries (three) will report available data on communicable diseases (49) as described in decision No. 2119/98/EC to the system. Apart from routine surveillance, TESSy will also replace the data collection for the previously-functioning DSNs, which are now gradually migrating from the contracted network hubs towards coordination of ECDC. This will provide European experts with a one-stop-shop for European surveillance data on communicable disease.

1.2 Document dependencies and other sources of information

This document is closely related to the TESSy Transport Protocol (XML) V2, which should be considered a prerequisite. That document describes the TESSy concepts, the TESSy data flow and how to compose a valid TESSy XML document.

To get a basic understanding of the technical concepts described in this document, see:

- XML: http://www.w3schools.com/xml/default.asp
- XML schema: http://www.w3schools.com/schema/default.asp
- Web service: http://www.w3schools.com/webservices/default.asp
- SOAP: http://www.w3schools.com/soap/default.asp
- XPath: http://www.w3schools.com/xpath/default.asp

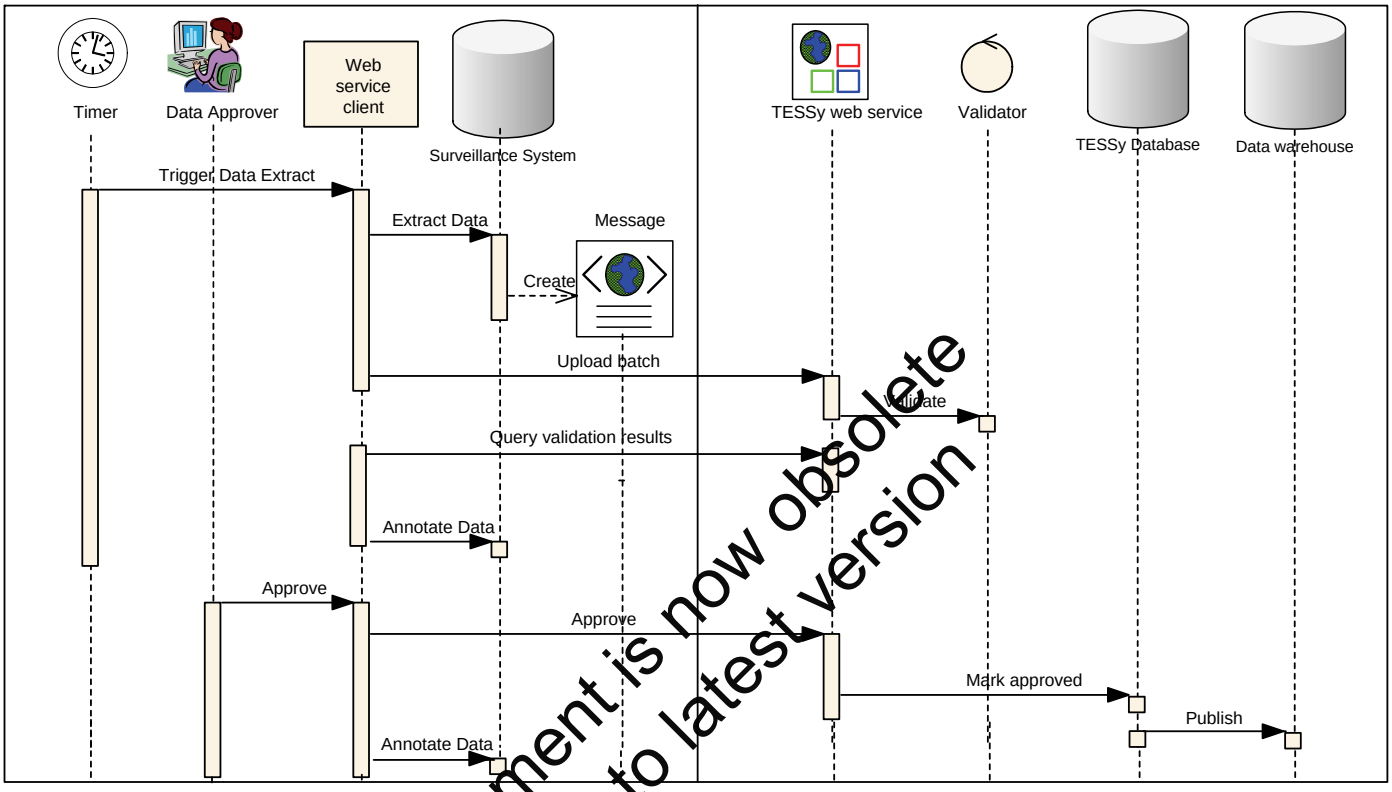
1.3 Glossary/Definitions

Batch	Contains a file with the information to report (Reporting Periods and Records) to the system. A batch is first uploaded by a user, who can later choose to approve or reject the batch. A batch in TESSy CSV consists of one file that contains the Records (Reporting Periods are added by the user in the web application during the upload action), while a batch in TESSy XML consists of one file that contains both Records and Reporting periods. Synonyms: Report batch, Data file.
Batch identifier	The batch identifier is only needed when using web services. It is an internal reference for each organisation to use in order to identify uploaded batches.

	The batch identifier is an incremental positive integer (starting at 1) that is given by the web service client. TESSy will enforce that each batch identifier given by the web service client is unique and greater than all previously used batch identifiers (see below), otherwise the batch will be automatically rejected. *batchId > max(previously used batchId:s)* It is recommended that a web service client increment the batch identifier by 1 as this will allow each organisation to discover missing batches (batches that have been sent by the web service client, but never received by TESSy). Hint: It is a good idea to use the identity column from the database table that keeps track of uploaded batches as the batch identifier.
Error	An error is a severe validation failure, which will cause the batch to be automatically rejected. Synonyms: Validation failure, Validation result.
Metadata	Defines TESSy data structures and contains: RecordTypes, Fields, Coded Value Lists, Coded Values, etc. The metadata also contains all validation rules related to Fields and RecordTypes. To achieve a maximum of flexibility, TESSy is very much metadata-driven, i.e., new Record Types can be added and Fields can be added, changed or removed without programming effort. These changes (and creation of each new instance of the metadata set) are documented in the TESSy help menu.
Record	An information item with a specified Record Type entered into TESSy, usually a case report or an aggregate entry.
Remark	A remark is used in the validation process to indicate an unlikely value or an unlikely combination of values. It serves for informational purpose and no immediate reaction by the user is required. Example: A five year-old boy notified as homosexual. Synonyms: Improbable data, Advice to change, Comment.
Reporting Period	Describes the intended availability of data on a subject in a specified timeframe for a Data Source. This information is important to distinguish reporting of zero cases of a disease from not reporting this disease at all during the specified time.
TESSy XML	A short term for the format specified in the document *Transport Protocol Specification, XML – Extensible Markup Language, TESSy*.
Warning	A warning is a minor validation issue. The user who approves the batch should review the warning and decide whether the dataset must be changed according to the warning message (reject the batch) or it should be confirmed unchanged, therefore overriding the warning messages (approve the batch). A warning can often set one or more Fields to unknown as a simple default data cleaning action. If any data cleaning will occur upon acceptance of the batch, the warning message will state this as well. Synonyms: Validation warning, Validation result.
Web Service	A standard protocol for machine-to-machine communication. TESSy provides a Web Service-based interface to upload data. Synonyms: Machine-to-machine connection.
XML	EXtensible Markup Language – A data transport format in which the data are structured hierarchically.

1.4 Machine-to-machine communication with TESSy

The following diagram describes how machine-to-machine communication with TESSy works:



1. At a certain point in time, either triggered by a manual action (like pressing a button) or a scheduled event, the web service client first extracts data from the national surveillance system and exports it in an XML file compliant with TESSy XML.
   It is recommended to automatically test the exported file against the TESSy XML schema.
2. The web service client then logs into the TESSy web service and uploads the exported file (batch) to the TESSy system. The web service client should process the response to identify any problem.
   It is highly recommended to first test the exported file using the appropriate TESSy web service method.
3. While the uploaded batch is being validated by TESSy, the web service client repeatedly queries the TESSy web service whether the validation has finished.
4. When the batch is validated, the web service client queries the TESSy web service for the validation results of the batch. The web service client should process the response to learn whether the batch was validated successfully or whether it was automatically rejected due to errors. Any error (and most warnings) must receive manual attention before restarting from step 1.
   If the batch was successfully validated and no errors were found during validation, then the corresponding data that was exported and reported must be annotated (see section 1.5) in the national surveillance system. It is recommended to annotate all records with their validation warnings or remarks for quality assurance.
5. Before the process of reporting is finalised, the validation results must be reviewed and the batch must be approved by an authorised person (user). This can be done in two ways:
   - Using the TESSy web service:
     – The validation results are processed and displayed for a user who will decide whether to approve the batch or reject it and restart from step 1 after addressing one or more validation messages.
     – If the decision was to reject the batch, the user should apply the appropriate method in the TESSy web service and then process the response to confirm that the rejection was successful.
       If the decision was to approve the batch, the user should apply the appropriate method in the TESSy web service and then process the response to confirm that the approval was successful.

- Using the TESSy web interface:
  - The user logs into the TESSy web interface and opens the validated batch in Review Uploads menu. The validation results should be carefully reviewed before the user decides to either approve or reject the batch.
  - If the decision is to reject the batch, then the user must click the button for rejection. If the decision is to approve the batch, then the user must click the button for approval.

6. If the batch is approved, then the corresponding data that was exported and reported must be annotated in the national surveillance system (see section 1.5). At the next time of export, it must be possible to distinguish which data records have already been reported to TESSy and which have been updated, deleted or newly inserted after.

1.5 Annotation in the national surveillance system database

In the previous subsection it is mentioned that the data in the national database should be annotated. By annotation we mean:

After validation:

- Since each error from the batch validation will prevent you to report this/these record(s) before the error has been addressed and fixed, you need to somehow keep track of which records had which errors. It does not matter whether you do this adding the information into the data model or prefer to keep the information in a log file. What is important is that you are able to identify which records had which problems in order to address the issues.
  It is recommended to do the same also for warnings and remarks. A validation message is generated for a reason, with the purpose of improving your data.

After approval:

- All the records (cases and/or aggregated records) included in the batch have now been reported to ECDC. If you are using the Update functionality, it is vital that you mark this information in the national database. The next time that you report data to ECDC you must be able to tell which records have been deleted and updated since the last submission and which records has not been reported. Here follows an example of how this can be solved:
  - Marking each record in the national database if and when it was last reported to ECDC. Each record should also have a last modified field and all records that are not yet reported to ECDC or have been modified since the last report should be included in the new batch. Also, do not forget to include all records that have been deleted from the local database since the last report. These should be reported with status delete.
- If your national database cannot comply with this functionality, you must use the Replace functionality for all data that you upload into TESSy. You might be forced to always include the full history for the disease. Please contact TESSy helpdesk to discuss how to ensure that you include sufficient information.
  The downside of always including the full history of the disease is that the number of validation messages tend to be proportional to the amount of uploaded data.

1.6 The journey to machine-to-machine communication

This document describes various aspects of machine-to-machine or web service communication with TESSy. The following list briefly describes the steps needed to acquire access, develop and test the national system and to implement the criteria list to request production access.

1. Read this document. Any questions will be answered by the TESSy team from tessy@ecdc.europa.eu.
2. Compete the *TESSy web service user account request form* listed in Appendix 2 and submit it to tessy@ecdc.europa.eu. You will receive user information validation in the test environment shortly after.
3. Start developing and testing a web service consumer that will be used to communicate with TESSy. The test application (section 2.3) can provide helpful guidance.
4. By this time, you should have developed an automated way of exporting your files to TESSy XML. Use the web service consumer to upload and test your exported files.
5. Write a standard operating procedure describing how the machine-to-machine communication is intended to be used and who is responsible for what (see section 6 for more information).
6. When ready, use the criteria checklist in Appendix 1 to make sure that all prerequisites are fulfilled. Submit a request for production use to tessy@ecdc.europa.eu.

7. You will receive access to the production environment when the request have been reviewed and approved by the TESSy team.
8. After the first use of the web service in production, a special check is performed to verify the success.

1.7 Support

ECDC offers support to Member States and TESSy users in several ways:

- Support – Support for TESSy data preparation, uploading and viewing is available via TESSy Helpdesk. The helpdesk helps the users with any access and errors in the system and with any data-specific answers. The helpdesk is available from 9am to 4:30pm CET (Stockholm time) on ECDC working days.
- Contact: TESSy Helpdesk: tessy@ecdc.europa.eu; +46 (0)8 5860 1601
- User training – To help users adjust to reporting data to TESSy, certain assistance and training will be available. Contact tessy@ecdc.europa.eu for details.

1.8 Document structure

This document is divided into six sections, with this introduction being the first. *Getting started* describes how to acquire user TESSy web service user credentials and includes a brief explanation of how to use the TESSy web service using java or C#. *Making requests* describes the method interface of the TESSy web service. *Processing responses* describes the XML schema that is defined for the TESSy web service method calls responses. *Using the web service in test* describes how to use the TESSy test environment, while the last section, *Using the web service in production*, describes the production environment as well as which criteria you must fulfil to start using the production TESSy web service.

Appendix 1 contains a checklist with the criteria listed in the section 6.

2 Getting started

This section describes how to start implementing your TESSy web service client. If you have any questions or requests, you are always welcome to contact the TESSy helpdesk.

2.1 Getting a user account to the TESSy web service

You can request a TESSy web service user account by filling in and sending the form *Request for access to The European Surveillance System – TESSy, TESSy web service user* (see Appendix 2). Note that you cannot use a regular TESSy user to utilise the TESSy web service, but must request separate user credentials for this.

The user account details that you receive back from TESSy Helpdesk will have full permissions for testing purposes. Later, when you are given access to the production system, the user permissions will be limited to the permission level indicated on the request form.

You are welcome to contact TESSy Helpdesk for questions or help on how to implement your TESSy web service client.

2.2 Implementing your TESSy web service access layer

The TESSy web service requests can be integrated into applications using the most modern programming languages.

2.2.1 Java

You can implement TESSy web service operations directly in Java. This section describes how to get started when using Java as the implementation language.

Generating the stubs

You can use the wsimport utility in Java 6 to generate the stubs from the TESSy web services WSDL (see 5.2 or 6.2 for WSDL location in test and production). If you are using previous Java versions, you can use Apache Axis.

To generate the TESSy web services client-side library stubs:

1. Go to the directory where you want to generate the stubs and create a “build” directory and a “src” directory.
2. All of the generated source code will go under the “src” folder.
   If you are using Eclipse 3.2 create a custom binding to disable “Wrapper Style” code generation.
   ```
   <jaxws:bindings wsdlLocation="
   https://tessy.ecdc.europa.eu/TessyV2TestWebService/TessyUpload.asmx?WSDL"
   xmlns:jaxws="http://java.sun.com/xml/ns/jaxws">
   <jaxws:enableWrapperStyle>false</jaxws:enableWrapperStyle>
   </jaxws:bindings>
   ```
   This step is necessary because Eclipse 3.2 does not support wrapper style generated code. However, if you are using an IDE that does support wrapper style generated code, such as NetBeans, this step is not required.
3. Run the command:
   ```
   wsimport -d ./build -s ./src  -p ecdc.tessy.webservice.client.jax
   https://tessy.ecdc.europa.eu/TessyV2TestWebService/TessyUpload.asmx?WSDL -b jaxws-
   custom.xml -extension
   ```
4. You can find the generated stubs in the path `ecdc.tessy.webservice.client.jax`.

Generated file types

Several file types are generated in the package `ecdc.tessy.webservice.client.jax`:

- Two classes for every method in TessyUpload, one to use for requests and one for the response.
- TessyUpload – Used to create the object implementation of the TESSy web service.
- TessyUploadSoap – Interface that the created object will implement.

Using the web service

You can now reference the SOAP proxy using your project namespaces. For example:

```
try {
    ecdc.tessy.webservice.client.jax.TessyUpload service =
```

```
                                    new ecdc.tessy.webservice.client.jax.TessyUpload();
    ecdc.tessy.webservice.client.jax.TessyUploadSoap port =
                                    service.getTessyUploadSoap();
    byte[] xml = GetFileToRawArray(fileName);
    ecdc.tessy.webservice.client.jax.TestDataResponse result = port.testData(xml);
    System.out.println("Result = " + result.getTestDataResult() );
} catch (Exception ex) {
    // TODO handle custom exceptions here
}
```

2.2.2 C#

You can implement TESSy web service operations directly in .Net. This section describes how to get started when using C# as the implementation language.

Create the SOAP proxy in Visual Studio

In your application, you need to add a web reference to the TESSy web services WSDL you want to use.

To add a web reference:

1. From the Project menu, select Add Web Reference.
   A dialog box opens.
2. Enter the WSDL URL for TESSy web services in the URL box.
   `https://tessy.ecdc.europa.eu/TessyV2TestWebService/TessyUpload.asmx?WSDL`
3. Enter the name of the web reference.
4. Click Go.
   The main pane in the dialog box shows the API.
5. Click Add Reference.
   A new Web References folder is added to the Solution Explorer.

Using the web service

You can now reference the SOAP proxy using your project namespaces. For example if the web reference name is TessyWS:

```
TessyWS.TessyUpload ws = new TessyWS.TessyUpload();
byte[] xml = GetFileToRawArray(fileName);
string result = ws.TestData(xml);
```

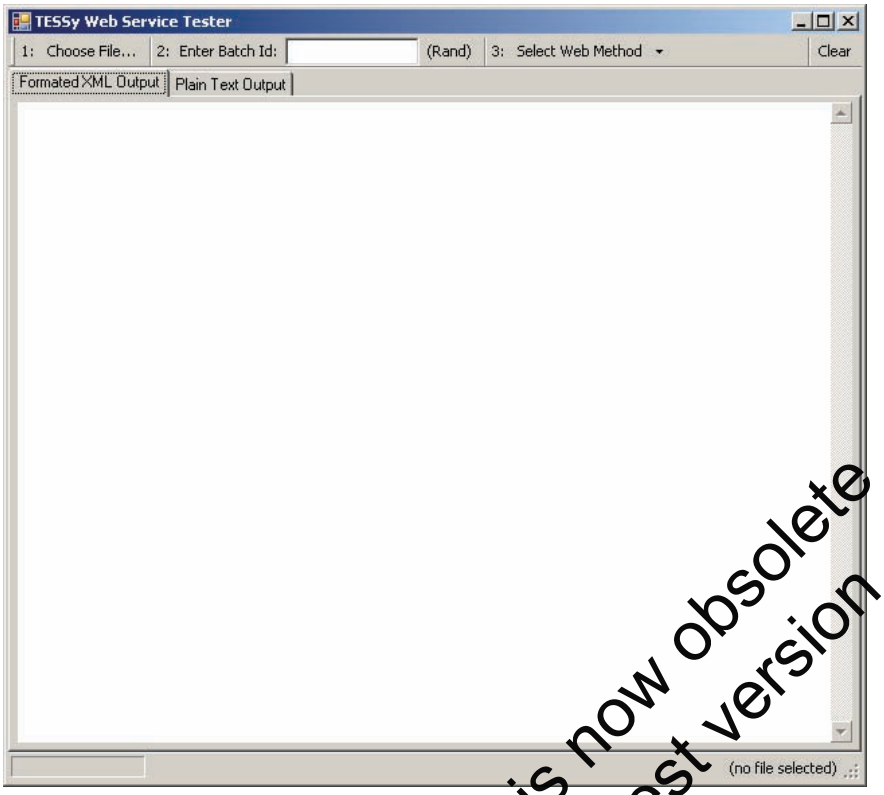

2.3 Test application (TESSy web service tester)

To further demonstrate how to use the TESSy web service using .Net 2 and C#, ECDC has developed a test application that suits two purposes. First, it serves as an example about how you can implement a TESSy web service consumer, and secondly, you can use it to easily test whether your TESSy username and password works successfully.

The application is called TESSy web service tester and the Visual Studio solution can be downloaded from http://tessy.ecdc.europa.eu/tools/TESSy web service tester.zip.

To test if your TESSy credentials work, follow these instructions:

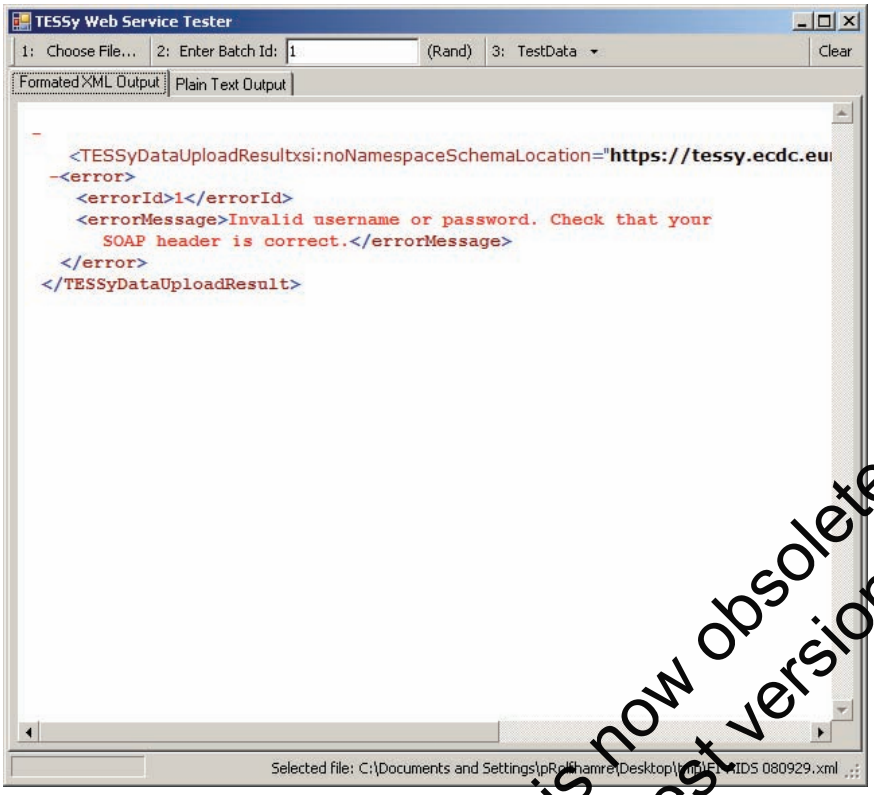
1. Download and open the solution in Visual Studio
2. Open Form1.cs (view as code) and scroll to the property `ServiceGatewayProxy`.
   Modify the following rows with your user name and password:
   ```
   ah.Username = "[user name]";
   ah.Password = "[password]";
   ```
3. Compile and run the application. This form should appear:



4. To test your credentials:

   a. Choose a TESSy XML file
   b. Enter a batch id, for example 1
   c. Click Select Web Method and select TestData
   d. Click TestData

The application will now test the selected file using the TestData method in the TESSy web service and will show the response in the output dialog. If you get the message below, please contact TESSy helpdesk to troubleshoot your user account.



3 Making requests

3.1 Authentication

When making method calls to the TESSy web service, you need to pass on the user name and password using as network credentials.

Example in C#:

```
tessyWS.TessyUpload client = new tessyWS.TessyUpload();
client.Credentials = new NetworkCredential(“[userName]”, “[password]”, “[domain]”);
```

Example in Java:

```
PasswordAuthentication auth = new PasswordAuthentication(“[domain]\[username]”,
                                                         “[password]”.toCharArray());

Set credentials = new HashSet();
credentials.add(auth);
connection.setCredentials(credentials);
```

3.2 Interface summary

This section lists all methods available in the TESSy web service. The functionality covered is basically the same as what is available through the web application, excluding data download and reports.

All methods will return XML strings. For more information on how to process the response, see section 4.

3.2.1 ApproveBatch

Parameter(s): int batchId – The id of the batch to approve

Returns: XML – operationSuccessful, operationNotSuccessful, noSuchBatch or error (see section 4)

Approve a batch. This method should only be used when the batch has been uploaded, validated and after carefully reviewing the validation results. Indicate the batch id when using the method to identify which batch you intend to approve.

The batch identifier is unique for each organisation and is provided when using the UploadData method.

3.2.2 AreValidationResultsAvailable

Parameter(s): int batchId – The id of the batch check if validation results are available

Returns: XML – areValidationResultsAvailable, noSuchBatch or error (see section 4)

Check if validation results are available. This method should be used to check if an uploaded file has been validated by the system. Use the GetValidationResults method to fetch the validation summary when the response agrees. Indicate the batch id when using the method to identify which batch you wish to investigate.

The batch identifier is unique for each organisation and is provided when using the UploadData method.

3.2.3 GetAllBatches

Parameter(s): None

Returns: XML – batchList or error (see section 4)

Fetch batch information for all batches that have been uploaded. This method should be used to get an overview of all batches that have been reported to the system. Use this method in combination with the GetValidationResults method to get more information about the uploaded batches if they have been validated.

3.2.4 GetBatchState

Parameter(s): int batchId – The id of the batch for which to fetch the batch state

Returns: XML – batchState, noSuchBatch or error (see section 4)

Get the batch state of an uploaded batch. This method should be used to lookup the batch state for an uploaded batch. Indicate the batch id when using the method to identify which batch you wish to learn the batch state for. Possible states are Uploaded, Failed, Validated, Approved, Rejected, RolledBack, Published.

The batch identifier is unique for each organisation and is provided when using the UploadData method.

3.2.5 GetLastBatchId

Parameter(s): None

Returns: XML – operationSuccessful or error (see section 4)

Get the last batch id uploaded by the current user's organisation. Each organisation using machine-to-machine communication with TESSy should keep track of which batches have been uploaded with their batch ids. This method should be used to identify whether the last reported batch corresponds with the last uploaded batch, otherwise indicating a problem that should be alerted to the system administrator for follow-up.

3.2.6 GetValidationResults

Parameter(s): int batchId – The id of the batch for which to fetch the validation results

Returns: XML – validationResults, noSuchBatch or error (see section 4)

Get the validation results for a batch. This method should be used to fetch the validation results for a batch that has been uploaded and validated. Indicate the batch id when using the method to indicate for which batch you wish to fetch the validation summary.

The batch identifier is unique for each organisation and is provided when using the UploadData method.

3.2.7 RejectBatch

Parameter(s): int batchId – The id of the batch to reject

Returns: XML – operationSuccessful, operationNotSuccessful, noSuchBatch or error (see section 4)

Reject a batch. This method should only be used when the batch has been uploaded and validated, and after the validation results have been carefully reviewed. Indicate the batch id when using the method to identify which batch you intent to reject.

The batch identifier is unique for each organisation and is provided when using the UploadData method.

3.2.8 TestData

Parameter(s): base64Binary xml – The byte stream of the TESSy XML file to test

Returns: XML – validationResults or error (see section 4)

Test a TESSy XML file before upload. This method should be used to test a TESSy XML compliant file before the file is uploaded in order to identify any validation problems. By testing the file before uploading it, you get immediate response and do not have to use the AreValidationResultsAvailable and GetValidationResults methods to retrieve the validation messages.

3.2.9 UploadData

Parameter(s):

- int batchId – The id of the batch to report.
- base64Binary xml – The byte stream of the TESSy XML file to report

Returns: XML – operationSuccessful, operationNotSuccessful or error (see section 4)

Upload a TESSy XML file to TESSy. This method should be used to report a TESSy XML compliant file to TESSy. Indicate the batch id when using this method, which is a positive integer uniquely identifying the batch being uploaded. The batch identifier must be greater than all previous batch identifiers from the same organisation. Otherwise, the system will know that one or more batches are missing.

4 Processing responses

All TESSy web service methods will return a text string that can be parsed as XML, compliant to an XML schema. The latest version of this XML schema can always be found here:

https://tessy.ecdc.europa.eu/schemas/TESSyDataUploadResultV2.xsd

This section will describe the XML schema and its elements.

4.1 Key elements of the XML schema

An XML schema is a description of a type of XML document, typically expressed in terms of constraints on the structure and content of documents of that type, above and beyond the basic syntax constraints imposed by XML itself. An XML schema provides a view of the document type at a relatively high level of abstraction.

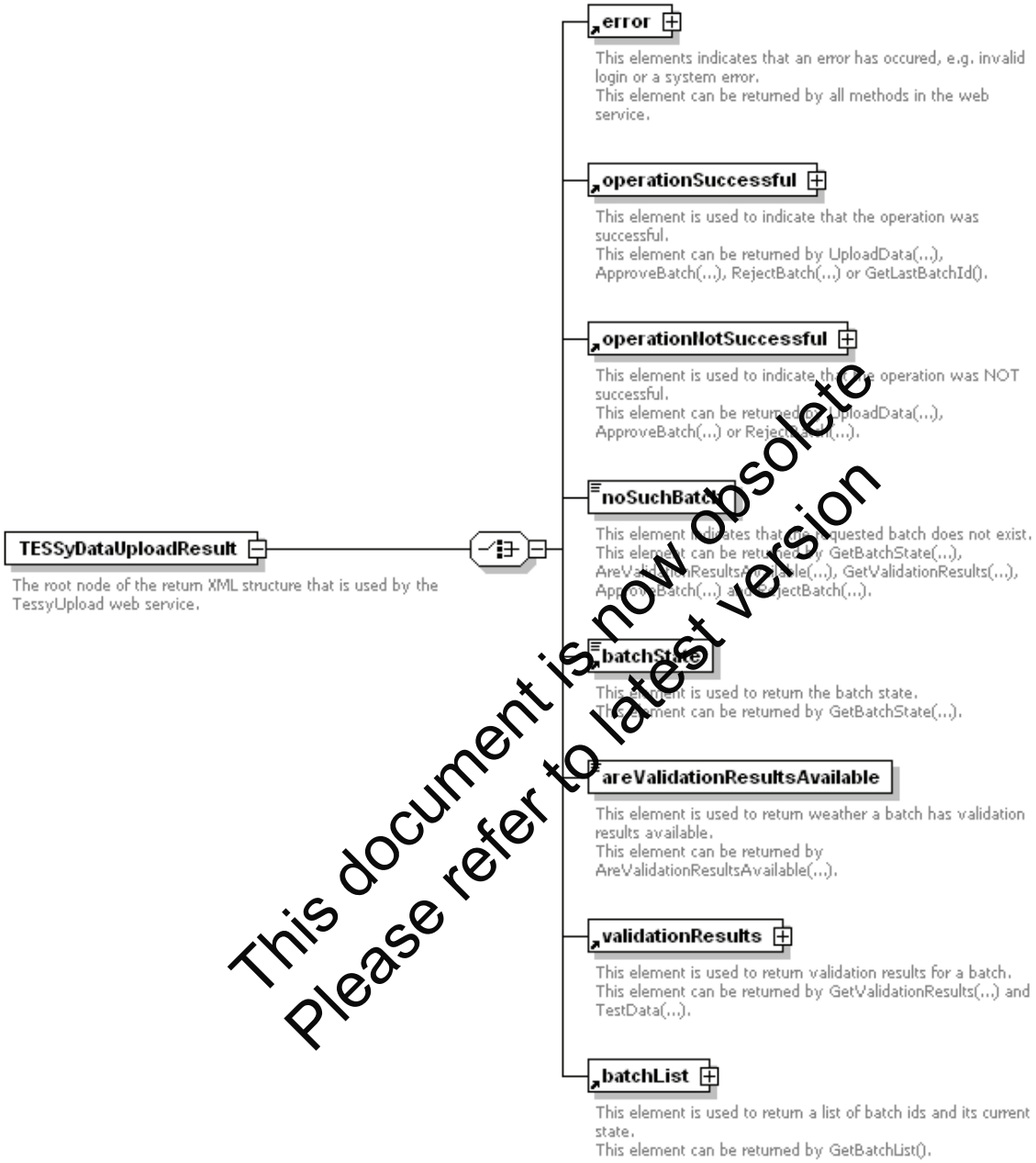
4.1.1 How to read the diagrams

The schematic presentation of the schema as used in this document uses several visual elements which will be explained here:

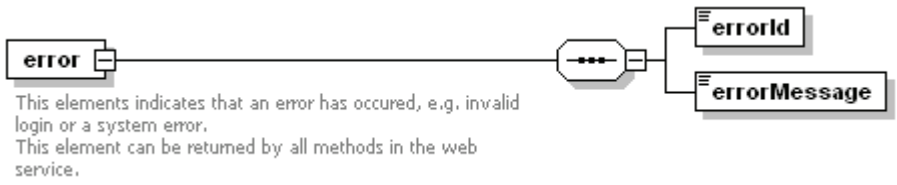
- **Box:** This is a XML data element, which contains data or other elements.
- **Box with dotted line:** Data element that is conditional (can be left out).
- **Plus sign:** Indicates that the element contains other elements.

Cardinality like 1..∞: Indicates the minimum and maximum number of occurrences of an element. If not indicated the maximum is always 1, the minimum can be derived from the solid or dotted line in which the element box is represented.

4.1.2 TESSyDataUploadResult



4.1.3 error



Type(s):

- errorId: xs:nonNegativeInteger
- errorMessage: xs:string

Example:

```
<error>
        <errorId>1</errorId>
        <errorMessage>
              Invalid username or password. Check that your SOAP header is
              correct.
       </errorMessage>
</error>
```

4.1.4 operationSuccessful

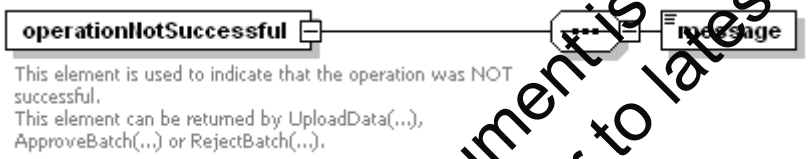
Type(s):

- batchId: xs:int

Example:

```
<operationNotSuccessful>
        <batchId>1</batchId>
</operationNotSuccessful>
```

4.1.5 operationNotSuccessful



Type(s):

- batchId: xs:int

Example:

```
<operationNotSuccessful>
        <message>No batch with batchId 1 was found.</message>
</operationNotSuccessful>
```

4.1.6 noSuchBatch

Example:

```
<noSuchBatch />
```

4.1.7 batchState

Type(s):

- batchState: { Uploaded, Failed, Validated, Approved, Rejected, RolledBack, Published }

Example:

```
<batchState>Uploaded</batchState>
```

4.1.8 areValidationResultsAvailable

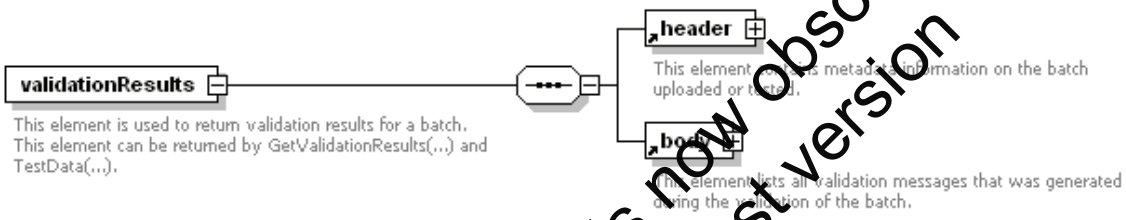
Type(s):

- areValidationResultsAvailable: xs:Boolean

Example:

```
<areValidationResultsAvailable>true</areValidationResultsAvailable>
```

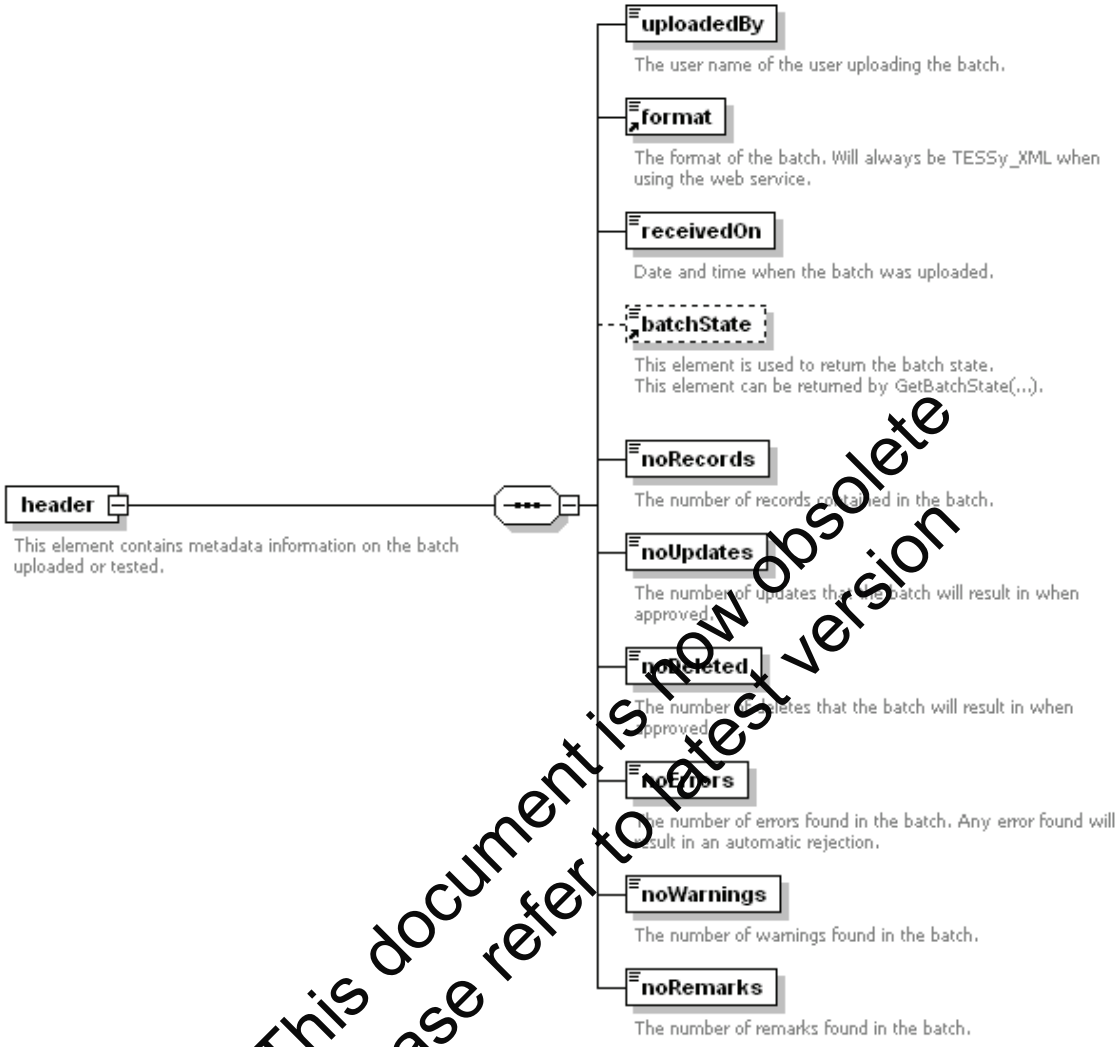
4.1.9 validationResult



Example:

```
<validationResults>
        <header>
                <uploadedBy>TestUser</uploadedBy>
                <format>TESSy_XML</format>
                <receivedOn>2008-10-09T20:06:57</receivedOn>
                <noRecords>2</noRecords>
                <noUpdates>0</noUpdates>
                <noDeleted>0</noDeleted>
                <noErrors>0</noErrors>
                <noWarnings>2</noWarnings>
                <noRemarks>0</noRemarks>
        </header>
        <body>
                <validationResult>
                        <type>warning</type>
                        <recordId>3827</recordId>
                        <subject>HIV</subject>
                        <message>
                           DateOfDiagnosis can not be later than DateOfNotification;
                           DateOfDiagnosis will be set to Unk
                    </message>
                </validationResult>
                <validationResult>
                        <type>warning</type>
                        <recordId>3828</recordId>
                        <subject>HIV</subject>
                        <message>
                           DateOfDiagnosis can not be later than DateOfNotification;
                           DateOfDiagnosis will be set to Unk
                    </message>
                </validationResult>
        </body>
</validationResults>
```

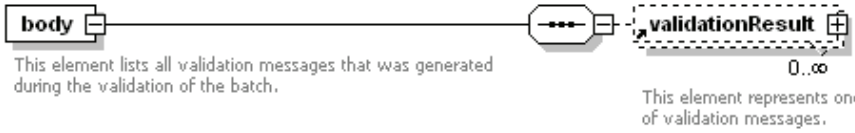
4.1.10 validationResult / header



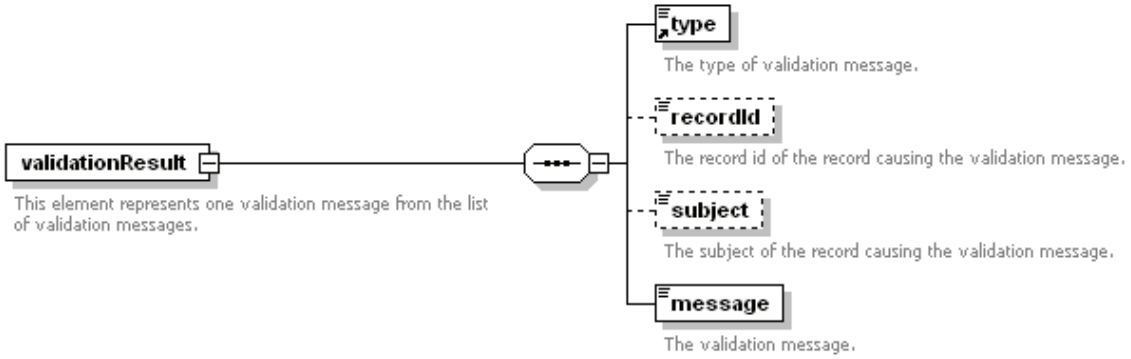
Type(s):

- uploadedby: xs:string
- format: { TESSy_XML }
- receivedOn: xs:dateTime
- batchState: { Uploaded, Failed, Validated, Approved, Rejected, RolledBack, Published }
- noRecords: xs:nonNegativeInteger
- noUpdates: xs:nonNegativeInteger
- noDeleted: xs:nonNegativeInteger
- noErrors: xs:nonNegativeInteger
- noWarnings: xs:nonNegativeInteger
- noRemarks: xs:nonNegativeInteger

4.1.11 validationResult / body



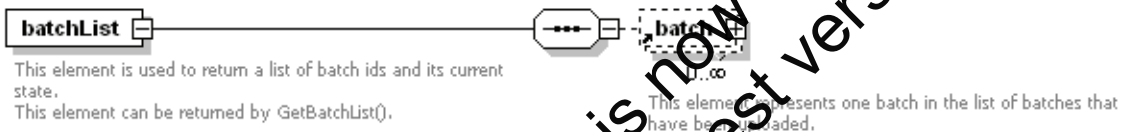
4.1.12 validationResult / body / validationResults



Type(s):

- type: { error, warning, remark }
- recordId: xs:string
- subject: xs:string
- message: xs:string

4.1.13 batchList



Example:

```
<batchList>
        <batch batchId="1" state="Rejected" />
        <batch batchId="2" state="Published" />
        <batch batchId="3" state="Validated" />
        <batch batchId="4" state="Uploaded" />
</batchList>
```


4.1.14 batch

Type(s):

- batchId: xs:int
- state: xs:string ({ Uploaded, Failed, Validated, Approved, Rejected, RolledBack, Published })

4.2 Errors

All errors are returned as an error code – error message combination. Currently, there is no predefined mapping between the error code and the error message. Although this is a clear requirement for the future, currently the error messages must be given manual and human attention when occurring.

5 Using the web service in test

This section describes the TESSy web service test environment and how it should be used.

5.1 Procedure to get access

The TESSy web service test environment is open for anyone to try out and use, the only requirement is a TESSy web service user account (see section 2.1).

Note that this is a test environment and that all submitted information might get deleted without further notice. Also, note that it is not recommended to send any sensitive information to the system.

5.2 Technical details

TESSy web service: https://tessy.ecdc.europa.eu/TessyV2TestWebService/TessyUpload.asmx

WSDL file: https://tessy.ecdc.europa.eu/TessyV2TestWebService/TessyUpload.asmx?WSDL

5.3 Support

The TESSy Helpdesk can answer questions regarding the system behaviour, why a specific batch has been rejected or any other question related to TESSy. They are also happy to provide guidance on how to implement a web service client and information about best practices when designing such a client.

Unfortunately, the TESSy Helpdesk cannot help with the implementation of a web service consumer. The TESSy web service tester (see section 2.3) gives a good starting point for such an application.

Please note that the user account details requested from TESSy helpdesk will have full permissions in the testing system for testing purposes.

6 Using the web service in production

This section describes the TESSy web service production environment and how it should be used.

6.1 Prerequisites to get access

This subsection describes what an organisation must fulfil in order to be awarded access to the TESSy web service production environment. Since machine-to-machine communication with TESSy is more or less automatic, ECDC requires a certain capacity and routine of monitoring before an organisation is allowed to use the TESSy web service.

- The requester must be able to present ten different TESSy XML compliant files that have been exported from the national surveillance system. The exported files should contain live data and more than 100 records. The ten files should represent different diseases and should cover diseases with enhanced surveillance.
- The requester must be able to show five different submissions (as described in section 1.4) in TESSy test environment.
- The requester must present a standard operating procedure (SOP) describing the following:
  - How the machine-to-machine communication is triggered.
  - How it is ensured that all intended data are reported (uploaded, reviewed, approved) and that no data records from the national surveillance system get left out of the export.
  - How missing batch ids are checked for (see section 1.3 – Batch identifier).
  - How automatic rejections are handled.
  - Who reviews the validation results and decides whether to approve or reject the batch.
  - Whether the approval process will be carried out using TESSy web service or TESSy web interface (see section 1.4)
  - How the data in the national surveillance system are annotated.

  The SOP should contain a brief summary (covering at least all above items) in English.
- The requester must nominate an additional alternate web service client administrator, who, together with the primary administrator, can be contacted for issues that arise. Both persons must have knowledge and permissions to temporarily shut down the web service client if requested.
- The requester must indicate an IP address or an IP address mask from where the web service client will be invoked.

Submit the full package of files and documents including the checklist found in Appendix 1 to TESSy Helpdesk. The submission will be reviewed by ECDC and clarifications may be requested.

When the request for access to the production environment is awarded:

- the TESSy web service user will be given permissions to the production environment as nominated;
- the indicated IP address or mask will be added to the list of allowed hosts for the production environment.

6.2 Technical details

TESSy web service: https://tessy.ecdc.europa.eu/TessyV2WebService/TessyUpload.asmx

WSDL file: https://tessy.ecdc.europa.eu/TessyV2WebService/TessyUpload.asmx?WSDL

Note that the TESSy web service production environment uses IP filtering. Therefore, you must request and be awarded access to the production environment before you can access the above URLs.

6.3 Support

When using machine-to-machine communication with TESSy, the system administrator of the TESSy web service client must regularly check the client log files and monitor if uploaded batches have been successfully reported. For any problem, question or request, please contact the TESSy Helpdesk.

The TESSy Helpdesk will monitor the logging of files and the system events and will contact the system administrator in case of problems or unclear events.

Appendix 1: TESSy web service production criteria – check list

Criteria	Check
Ten different TESSy XML compliant files that have been exported from the national surveillance system. The exported files should contain live data and contain more than 100 records. The ten files should represent different diseases and should cover diseases with enhanced surveillance.	☐
Five different submissions (as described in section 1.4) in the TESSy test environment.	☐
A standard operating procedure (SOP) describing the following:	
a. How is the machine-to-machine communication triggered?	☐
b. How is it ensured that all intended data are reported (uploaded, reviewed, approved) and that no data records from the national surveillance system gets left out of the export?	☐
c. How do you check for missing batch ids (see section 1.3 – Batch identifier)?	☐
d. How is automatic rejections handled?	☐
e. Who reviews the validation results and decides whether to approve or reject the batch?	☐
f. Will the approval process be carried out using TESSy web service or TESSy web interface (see section 1.4)?	☐
g. How is the data in the national surveillance system annotated?	☐
h. The SOP should contain a brief summary (covering at least all above items) in English.	☐
A web service client administrator and an alternate nominated.	☐
IP address or an IP address mask from where the web service client will be invoked.	☐

Appendix 2: TESSy web service user account request form

Request for access to The European Surveillance System – TESSy

TESSy web service user

In order to receive user credentials (user name and password) to The European Surveillance System (TESSy) web service, the Request for Access to The European Surveillance System must be completed, signed by the officially designated responsible person for nominating TESSy users and sent to:

European Centre for Disease Prevention and Control – TESSy
171 83 Stockholm
SWEDEN
Fax: +46 8 58 60 1294

ECDC will send the TESSy web service user name and password to the email address you provide below.

I request a ***TESSy web service user account***[1] *with the permissions listed below, for the following institution*. All information is mandatory.

CONTACT INFORMATION

Institution: ……..

Address: ……

……..

Country: ……..

Web service main contact person[2]:

First name: ……..Last name:…….

Telephone: ……...Mobile: ……

Fax: ……E-mail:……..

[1] The web service user account will have full permissions in the testing system for testing purposes. Later, when you are given access to the production system, the user permissions will be limited to the permission level indicated on the request form. See chapters 4 and 5 in the TESSy Web Service protocol for detailed procedure on obtaining access to TESSy production system.

[2] The indicated contact person must have the ability to temporarily shut down the web service client. With request for access to the TESSy production system, an additional alternate contact point must be given to ECDC, who should also have the ability to manage the web service client.

DATA ACCESS

The web service user account should have data access to the following diseases:

☐ All diseases			
☐ AIDS/HIV	☐ Gonococcal infections	☐ Plague	☐ Trichinellosis
☐ Anthrax	☐ Hepatitis A	☐ Pneumococcal Infections	☐ Tuberculosis
☐ Botulism	☐ Hepatitis B	☐ Poliomyelitis	☐ Tularemia
☐ Brucellosis	☐ Hepatitis C	☐ Q-fever	☐ Typhoid/ paratyphoid fever
☐ Campylobacteriosis	☐ Inf. with H. infl.	☐ Rabies	☐ vCJD
☐ Chlamydia infections	☐ Influenza	☐ Rubella	☐ West Nile fever
☐ Cholera	☐ Legionellosis	☐ Salmonellosis	☐ Yellow fever
☐ Cryptosporidiosis	☐ Leptospirosis	☐ SARS	☐ Yersinosis
☐ Diphtheria	☐ Listeriosis	☐ Shigellosis	☐ AMR
☐ Echinococcosis	☐ Malaria	☐ Smallpox	☐ Health care assoc. diseases
☐ Infection with Enterohaemorrhagic E. coli (VTEC)	☐ Measles	☐ Syphilis	
	☐ Meningococcal disease	☐ Tetanus	
	☐ Mumps	☐ Toxoplasmosis	
☐ Giardiasis	☐ Pertussis		

The web service user account should have the following data access roles[3] (please mark all that apply):

☐ data uploader ☐ data approver

I am aware that the credentials (user name and password) to web service users must not be shared with others outside the administrator's team.

Signature of main web service contact:……Date:……..…..

Name of national nominator of TESSy users[4]: ……

Signature of national nominator (mandatory): ……....Date:……..

Data Protection Clause

Any personal data collected by ECDC will be processed solely for the purposes of the performance, management and follow-up of ECDC activities in the framework of which it is collected and in accordance with Regulation (EC) n. 45/2001 on the protection of individuals with regard to the processing of personal data. This may involve distribution of data to other Community institutions and bodies. Your data may be recorded and stored for no longer than is necessary for the purposes for which they are collected and, in any case, only as long as follow-up action is needed. You have the right to access and rectify your personal data at any moment. Should you have any queries or request concerning the processing of your personal data, please address the relevant Data Controller.

You have right of recourse at any time to the European Data Protection Supervisor.

For more information on personal data protection and related documents see the Personal Data Protection section of the Important Legal Notice on ECDC website.

[3] The web service client can only upload and approve data in the TESSy system. Please note that the approval of data can also be performed via the TESSy website. For further description, please see Chapter 1.4 in the TESSy Web Service Protocol.

[4] Main national epidemiological contact point for the MS.