Submitted to the EC on XXX/XX/20XX

Justice Action Grant

EXEC

Electronic Xchange of e-Evidences with e-CODEX

Call identifier: JUST-JACC-EJU-AG-2017
Project full title: Electronic Xchange of e-Evidences with e-CODEX
Grant agreement n°: 785818

D4.1: Testing Tools and Strategy

Deliverable Id:	D4.1
Deliverable Name:	Testing Tools and Strategy
Status:	Final
Dissemination Level:	Consortium
Due date of deliverable:	30/11/2018
Actual submission date:	23/01/2019
Work Package:	WP4
Organisation name of lead partner for this deliverable:	AUTH
Author(s):	Ioannis Pagkalos Zoi Kolitsi
Partner(s) contributing:	-

Abstract:
This deliverable outlines the key components of testing within the EXEC project: the Testing tools and the Testing strategy that will be followed until the end of the project. The document presents the Test schemas, the updated testing monitoring components in the e-CODEX Central Testing Platform and describes the high-level strategy for testing.

History

Version	Date	Changes made	Modified by
0.1	25/11/2018	First Draft	Ioannis Pagkalos Zoi Kolitsi
0.2	18/12/2018	Second draft Circulated to Consortium	Ioannis Pagkalos Zoi Kolitsi
0.3	16/01/2019	Consolidated consortium comments	Ioannis Pagkalos
1.0	23/01/2019	Final version (no changes)	Mathias Maurer

Table of contents

List of Figures

List of Tables

List of Abbreviations

Acronym	Explanation
CTP	Central Testing Platform
CTR	Common Test Reporting
E2E	End-to-End
EIO	European Investigation Order
EPO	European Payment Order
ESC	European Small Claims
GUI	Graphical User Interface
GW	Gateway
MS	Member State(s)
OASIS	Organization for the Advancement of Structured Information Standards
TA	Test Assertion
TAML	Test Assertion Markup Language
XML	eXtensible Markup Language

***Table 1**: Abbreviations*

Executive Summary

EXEC enables the participating Member States to exchange European Investigation Orders (EIO) and related e-Evidences fully electronically through the Reference Implementation provided by the European Commission or existing national back end solutions. The e-CODEX Building Blocks (DOMIBUS Gateway and Connector) build the exchange infrastructure that facilitates the electronic delivery of EIO and e-Evidences.

The objective of this document is to describe the testing tools and strategy that will aid in reaching the goal of exchanging e-Evidences via e-CODEX during the course of the EXEC project. At the time of submission of this deliverable, most member states are finalising the setup of their national e-CODEX infrastructure. EXEC WP4 has prepared the toolset required for testing in order to make it available before the year-long round of testing between EXEC project participants.

The sections that follow present this toolset, comprised of (i) the EXEC Test Assertion & Test Reporting schemas and (ii) the updated test monitoring plugin in the e-CODEX Central Testing Platform. This document also describes the principles of the testing strategy, setting the directions under which this new testing tools will be exploited.

The next steps in regard to testing and EXEC WP4 are to create, share and schedule tests in collaboration with EXEC partners, in order to begin testing according to the project timeline.

1 Introduction

Testing is a key element of any interoperability-driven project and has been an integral part of all large-scale pilots aiming to demonstrate feasibility and establish European cross-border connectivity in many areas, including e-Justice. In EXEC, testing the exchange of e-Evidences between prosecutors and their staff is a powerful tool to show, learn and evaluate the added value of the solution developed by the European Commission, e-CODEX and this consortium.

A rigorous End-to-End testing methodology before "going-live" will also allow the involved actors to check and establish cross-border connectivity as well as certify a member state's conformance to the e-Evidence standards, the correct semantic mapping between connector/implementation/backend and the overall successful e-Delivery in End-to-End communication.

However, the testing of all possible connections between Member States can be time consuming. In order to be able to efficiently test the exchange of e-Evidences and stay within the timeline of the "e-Evidence: the way forward" guidelines, testing in EXEC will follow a pairs-testing strategy that takes advantage of an XML-based testing toolset, "jump-started" by the *e-CODEX Central Testing Platform.*

1.1 The e-CODEX CTP

The e-CODEX Central Testing Platform (hereby referred to as: 'CTP') was developed during the lifetime of the e-CODEX project by the Aristotle University of Thessaloniki (AUTH), Greece – which also is the WP4 Leader in EXEC. It is an automated testing infrastructure that enables member states to perform End-to-End tests between their e-CODEX e-Delivery infrastructure and a fixed central testing point that is available 24/7 via a Web GUI. The CTP is not an interoperability testing platform, in the strict sense of testing adherence to a standard, but rather an easy-to-use visual tool for Member States' national administrations and other user communities to verify their successful national or use-case-specific deployment of the e-CODEX assets using a set of sample-based, customisable tests. Over its nearly 5 years of operation, the CTP has proven to be *highly successful in reducing the resources needed in e-CODEX for testing.*

Within the EXEC project, the CTP has been upgraded to support the future exchange of standard European Investigation Order forms and their attachments / e-Evidence. In addition, it now also supports the monitoring and coordination of tests through a central web interface. These new functions of the CTP, along with its basic operations, are described in the detail in Chapter 3 of this document.

WP4, which is also the entity responsible for monitoring, coordination and test reporting within EXEC, will ensure the updating, availability and maintenance of the CTP for the duration of the EXEC project.

1.2 Testing Strategy – Basic Principles

1.2.1 Objectives to be supported

Testing in EXEC shall support the following objectives:

(i) Allow Member States (MS) to test the proper function of their national e-Evidence implementation, identify areas needing improvements and successfully complete their solution

(ii) Allow MS to verify, in real-life conditions and using test data, their connectivity with other MS over the complete workflow of sending and receiving e-Evidences across borders

(iii) Create confidence with the end-users on the use of the tools, by providing evidence of appropriate functionality, usability and security in real life conditions

(iv) Monitor Progress per MS

1.2.2 Outcomes

(i) Readiness to enter E2E testing: Successful completion of a first round of automated testing will establish the readiness to enter E2E testing efforts.

(ii) Readiness to Go Live: Successful completion of E2E testing will establish the readiness of a MS to go live with other MS

(iii) Identify points of specific attention for Roll-out in a Member State

1.2.3 Testing Process

(i) Using the CTP as a central coordination point for testing, member states will first perform a specially-designed series of **semi-automated tests**, with the CTP as the role of receiver or sender, which will establish a baseline for future testing efforts. These tests will be designed by WP4 and presented in D4.2.

(ii) After having successfully verified their national infrastructure against the CTP, member states will then perform **End-to-End testing** in pairs, based on validated e-CODEX procedures, modified accordingly for e-Evidence.

These procedures are further described in Chapter 4.

EXEC Test reports, the results of these test scenarios per member state, use a *common, XML-based format* (described in Chapter 2) and will be uploaded to the CTP and subsequently shared within the consortium in order to track project-wide progress. To this effect, the CTP has been upgraded to provide simple Web Interfaces to generate these test reports (also presented herein).

Finally, the CTP now also includes a test monitoring interface that will allow relevant stakeholders (e.g. WP Leaders as well as the member states themselves) to historically track, analyse and consult the tests performed. For example, a MS is able to log-in to the CTP and see which tests have been assigned, when is their due date etc.

The sections that follow present the EXEC Test Assertion and Test Reporting schemas, as well as the updated components of the e-CODEX Central Testing Platform. This document also describes a best-effort, given the information available, high-level strategy for testing that will take advantage of this new testing toolset.

1.2.4 Advice for the speedy reader

This document describes multiple concepts and processes related to e-CODEX and EXEC testing:

Chapter 2 discusses the XSD schemas used to describe and report on tests.

- You should read this chapter if you're interested in learning more about how these XML files are structured.
- If you will perform tests within EXEC, you may initially skip this chapter as the CTP will automatically generate these XML files when testing through a simple Web UI.

Chapter 3 presents the e-CODEX Central Testing Platform and how it works

- If you will perform tests within EXEC, we advise that you consult this chapter.
- If you're already familiar with the CTP and would like to learn only about the new features, skip to Chapter 3.3

Chapter 4 presents a basic overview of the EXEC Testing strategy

- If you will perform tests within EXEC, we advise that you consult this chapter

2 EXEC Testing Tools – XSD Schemas

In order to harmonise the available tests within EXEC, as well as align with the general architectural choices in e-CODEX, EXEC Testing uses two XSD schemas for describing: **(1) test assertions** (or test scenarios) and **(2) test reports**, respectively. The idea behind this approach is:

- To make test assertions structured, shareable and machine-understandable, which promotes reusability
- To make test reports structured and able to be filled in via a Web GUI. This will also allow them to be easily collected and processed by a central testing coordinator

Historically, e-CODEX tests have been described in Excel sheets, where their progress and results were also tracked. Even though this approach has worked very well in the past, a more structured approach, coupled with appropriate Web interfaces in a central environment (e.g. the CTP), is expected to help reduce the effort in documenting, sharing and – subsequently – collecting and analysing tests.

2.1 EXEC Test Assertions Schema

When it comes to schema design, implementing existing, validated assets in a new solution - where possible - not only reduces the effort required for developing the solution but also promotes re-usability within the e-Justice domain. For example, the SBDH schema[1] has been used in e-CODEX forms since the project's start and has harmonised the handling of "standard" document metadata (e.g. uid, author, date, version etc.) between member states and their respective national backends.

In line with this thinking, EXEC test assertions re-use the OASIS TAML (Test Assertion Markup Language) XSD Schema to describe tests. TAML is a well-known, validated, OASIS standard that was also partly used in e-SENS and many other research, academic and enterprise environments. It describes test assertions in an abstract but well-structured format, which is a good fit for describing EXEC test assertions as well.

The section that follows briefly describes the basic architecture of OASIS TAML. More information and detailed Schema documentation can be found in the OASIS Test Assertions Guidelines:
[https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tag]

[1] Standard Business Document Header: [https://www.gs1.org/standards/edi/standard-business-document-header-sbdh]

2.1.1 OASIS TAML

OASIS TAML is a Markup Language that allows easy, structured expression of a Test Assertion (TA), which can be defined as “a testable or measurable expression for evaluating the adherence of an implementation (or part of it) to a normative statement in a specification[2]”

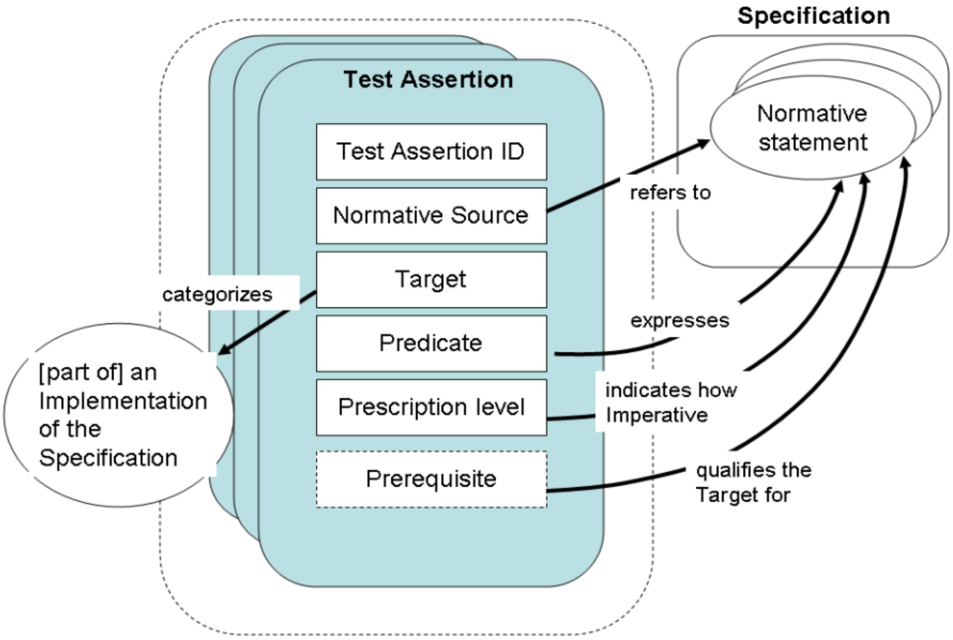


Figure 1: General Anatomy of a Test Assertion

As shown in **Figure 1** (sourced from OASIS TA Guidelines v1.0), a series of elements are defined with the OASIS TAML specification in order to describe multiple aspects of a TA, namely:

- **Test Assertion ID:** A unique identifier of the test assertion. It facilitates the mapping of assertions to specification statements. It is recommended that the identifier be made universally unique.
- **Normative Source**: The part of the test assertion that identifies the precise specification requirements or normative statements that the test assertion is addressing.
- **Target**: The target – or test target – categorizes an implementation or a part of an implementation of the referred specification that is the main object of the test assertion and of its Normative Source.
- **Predicate**: The predicate is a logical expression that asserts, in a testable form, the feature (a behaviour or a property) described in or referenced by the Normative Source, concerning

[2] As defined in OASIS TAG v1.0.pdf

an instance of Target. The Predicate evaluates to either “true” or “false”. If the predicate evaluates to “true” over the test assertion Target, this means that the Target exhibits this feature. “False” means the Target does not exhibit this feature.

The following terms are optional, in terms of the specification, but are very useful in fully describing a TA, and thus will also be included in EXEC TAs:

- **Description:** An informal definition of the role of the test assertion, with some optional details on some of its parts. This description must not alter the general meaning of the test assertion and its parts. This description may be used to annotate the test assertion with any information useful to its understanding. It does not need to be an exhaustive description of the test assertion.
- **Prescription Level**: A label in a test assertion that states how imperative it is for a Target instance to satisfy the Predicate condition. Three levels are defined in this specification: *mandatory* (corresponding to normative keywords MUST [NOT] / REQUIRED / SHALL [NOT]), *permitted* (MAY, OPTIONAL) or *preferred* (SHOULD [NOT] / RECOMMENDED).
- **Prerequisite:** The test assertion Prerequisite is a logical expression (similar to a Predicate) which further qualifies the Target for undergoing the core test expressed by the Predicate). It may include references to the outcome of other test assertions. If the Prerequisite evaluates to "false" then the Target instance is not qualified for evaluation by the Predicate. If the Prerequisite evaluates to "true" then the Target instance is qualified for evaluation by the Predicate. Prerequisites have also been called “preconditions” in [TMD].
- **Tag(s):** A Test assertion may be assigned 'tags'. These tags provide an opportunity to categorize the test assertions. They enable the grouping of test assertions, for example based on the type of test they assume or based on their target properties. Tags may be any string – e.g. some predefined 'keywords', which may in turn be given values.
- **Variable(s)**: Parameters or symbols employed when writing a test assertion used to refer to values that are not known or fixed at the time the test assertion is written, but will be determined at some later stage, possibly as late as the middle of running a set of tests. A variable is also employed to enable several assertions to share a value (set once, used by many), like a variable in other technologies.

2.1.2 EXEC Test Assertion Example

The following code snippet provides a simple example of an EXEC Test Assertion, using the OASIS TAML XSD.

Even though many of the values used are placeholders, the logic on how to use TAML is adequately presented. Based on a certain pre-approved requirement specification (e.g. “EXEC Consortium Test Assertion 01”), a Sender and a Receiver are configured to exchange e-CODEX message. The expected result is that the receiver successfully sends back an AS4 non-repudiation receipt.

```xml
<?xml version="1.0" encoding="UTF-8"?>

<testAssertion xmlns="http://docs.oasis-open.org/ns/tag/taml-201002/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://docs.oasis-open.org/ns/tag/taml-201002/
testAssertionMarkupLanguage.xsd" id="EXEC_TA_01">
    <description>THE DESCRIPTION OF THE TEST</description>
    <normativeSource>
        <refSourceItem>EXEC Consortium Test Assertion 01</refSourceItem>
    </normativeSource>
    <target>Connectivity</target>
    <prerequisite>Sender and Receiver are configured to exchange e-CODEX
messages</prerequisite>
    <predicate>The Receiver sends back an AS4 non-repudiation receipt</predicate>
    <prescription level="mandatory"></prescription>
    <tag tname="Type">Connectivity</tag>
    <tag tname="Group">EXEC</tag>
</testAssertion>
```

2.2 EXEC Test Reports Schema

With a solid basis in the OASIS TAML schema for test assertions, WP4 has developed a simple but fit-for-purpose XSD Schema to describe test reports for EXEC, named **“EXEC Common Test Reporting” (CTR)**. This includes basic elements to describe the test participants (both the author of the test and the sender/receiver pairs) as well as basic indicators of test results (success/failure) and – if applicable – error messages, non-repudiation evidences, and e-CODEX connector logs.

The full schema of the EXEC CTR can be found online at: [https://ecodextest.ee.auth.gr/ctr]. The section that follows briefly describes the basic elements of the schema and provide an example of its use.

2.2.1 The EXEC Common Test Reporting XSD

The EXEC Common Test Reporting (CTR) schema contains a basic element, the testReport, which is further described by entities and data that were part of the test. Most of these elements are abstractly defined (i.e. using text values instead of pre-determined lists where possible) in order to not restrict their application:

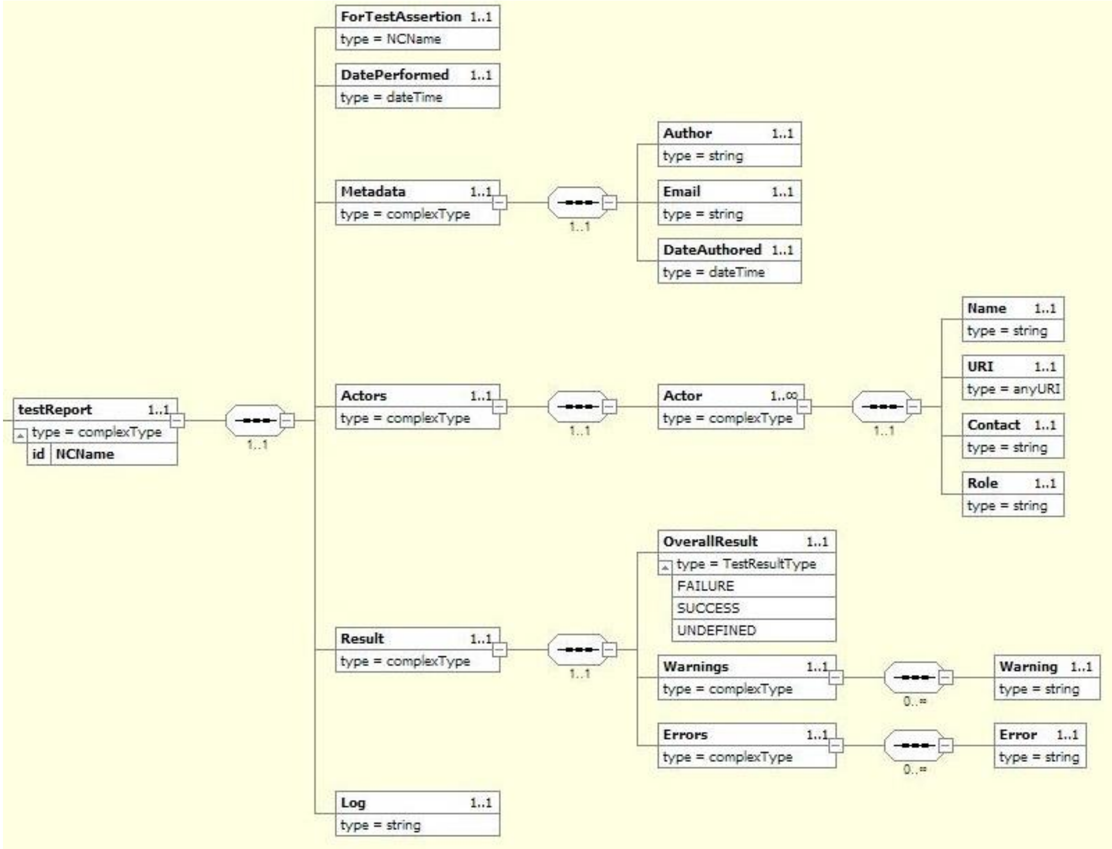


Figure 2: EXEC Common Test Reporting Schema

- **testReport id:** Uniquely identifies this test report
- **ForTestAssertion**: This field links the report to a Test Assertion (e.g. EXEC_TA_01)
- **DatePerformed**: Defines when the test was performed (not when the report was generated, this value is defined below)
- **Metadata** (**Author**, **Email**, **DateAuthored**): This set of values defines the contact details of the report's author (name, email) as well as the authoring date.
- **Actors**: This is a complexType element that defines a series of actors that took part in the test, further defined by their **Name**, **URI**, **Contact** (e.g. an email) and **Role** in the test (e.g. SENDER or RECEIVER)
- **Result:** This includes the **OverallResult** (predetermined choice between SUCCESS/FAILURE and UNDEFINED) value as well as an element to separately record **Warnings** and **Errors**
- Finally, a **Log** element is included in case server/application logs are to be included.

In summary, this is an abstract but structured way of defining the basic characteristics of each test report, with a direct link to the related assertion. Even though the assertion itself could be fully integrated in the test report XML as well, the same effect can be achieved on a visual/interface or application level by merging the two XML files.

2.2.2 Example of a CTR

In the following example, a successful test report is produced for EXEC_TA_01 (a sample EXEC Test Assertion, as described in 2.1.2).

Two actors, “CTP Domibus 2” and “GR TEST Domibus” have successfully exchanged messages and log the results of the test:

```
<?xml version="1.0" encoding="UTF-8"?>
<testReportContainer xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation=" http://wwww.e-sens.eu/CommonTestReport
EXEC_CommonTestReport.xsd"
    xmlns="http://wwww.e-codex.eu/EXEC_CommonTestReport"
    xmlns:taml="http://docs.oasis-open.org/ns/tag/taml-201002/">
    <testReport id="TestReport_2018-11-20_01">
        <ForTestAssertion>EXEC_TA_01</ForTestAssertion>
        <DatePerformed>2018-11-19T15:00:00+02:00</DatePerformed>
        <Metadata>
            <Author>Ion Pagkalos</Author>
            <Email>ipagkalo@auth.gr</Email>
            <DateAuthored>2018-11-20T09:00:00+02:00</DateAuthored>
        </Metadata>
        <Actors>
            <Actor>
                <Name>CTP Domibus 3</Name>
                <URI>http://1.2.3.4:8000</URI>
                <Contact>systemadmin@domibus1.gr</Contact>
                       <Role>SENDER</Role>
            </Actor>
            <Actor>
                <Name>GR TEST Domibus</Name>
                <URI>http://4.5.6.7:8000</URI>
                <Contact>systemadmin@domibus2.gr</Contact>
                       <Role>RECEIVER</Role>
            </Actor>
        </Actors>
        <Result>
            <OverallResult>SUCCESS</OverallResult>
            <Warnings/>
            <Errors/>
        </Result>

        <Log>CTP/Domibus Log</Log>
    </testReport>
</testReportContainer>
```

3 EXEC Testing Tools – the e-CODEX CTP

The e-CODEX Central Testing Platform (CTP) is a central platform for test automation, serving all existing and future e-CODEX piloting countries and use cases. Its main goal is to allow users to send and receive customisable e-CODEX messages without the need to involve another "testing partner". The CTP is part of an e-CODEX test environment configuration and is able to communicate with other e-CODEX members' infrastructure in order to aid in setting up and testing an e-CODEX installation. Technically, it consists of an e-CODEX gateway and an associated Web Graphical User Interface (Web GUI) that can be used to send messages to a partner's gateway as well as view messages that are sent to the CTP by the same gateway. The "original" CTP, developed by GR AUTH and used for testing EPO, Small Claims and other e-CODEX Pilots is maintained by the Me-CODEX project and serves the e-CODEX Test environment.

For the EXEC project, a new, separate instance of the CTP has been set up at https://exectest.ee.auth.gr which has been extended with specifically built-for-purpose Test monitoring and reporting tools that are discussed in the chapters that follow. This CTP instance will exclusively serve the EXEC Testing environment.

Even though the focus of this deliverable in regard to the CTP is to present the EXEC-specific functionalities, this section begins with a brief overview of the basic functionality provided by the e-CODEX CTP for the purposes of document self-sufficiency[3].

3.1 Testing Scenarios supported by the CTP

The CTP supports two distinct scenarios:

1. Connectivity Testing

Using the *GW-Test* and *Connector-TEST* message, the CTP can be used to perform connectivity tests between the CTP's and a member's e-CODEX infrastructure. This can be done automatically and at any time, by "ordering" a message to be sent to a member's gateway through the e-CODEX CTP Web

[3] More detailed documentation on these topics can be obtained via ecodex-ctp@auth.gr

GUI as well as sending a message to the CTP and checking its progress through the same interface. Doing so can save a great deal of time as it can effectively replace the need for manual Gateway-to-Gateway (GW2GW) connectivity testing which would typically involve coordination between two different tech teams and introduce delays caused by conflicting/busy schedules.

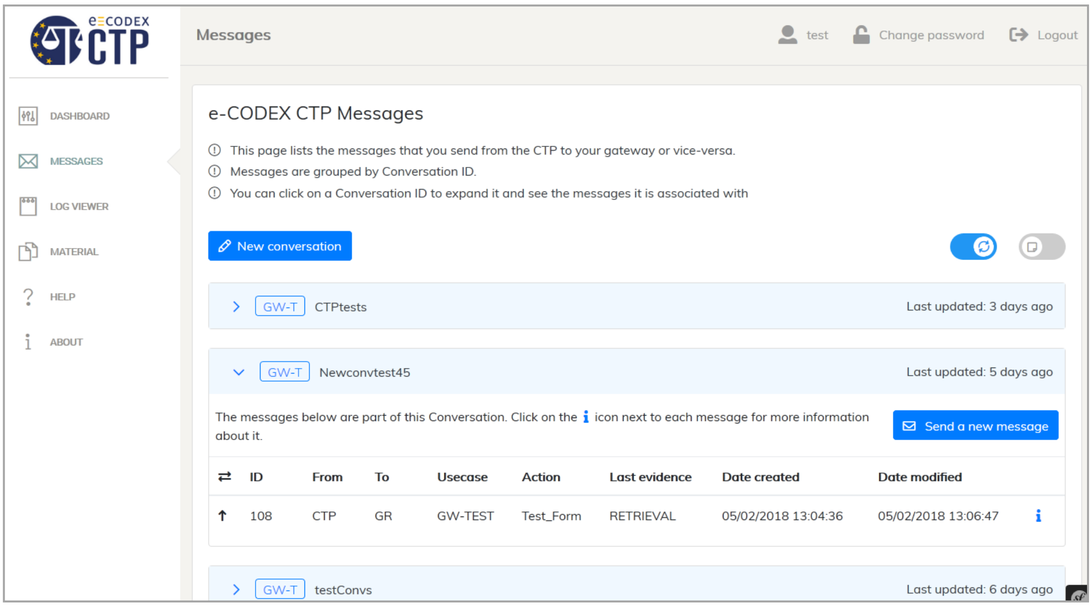
2. e-CODEX Use-case Testing

A user of the CTP can select one or more sample messages for *an e-CODEX use case* (in this case: Sample EIO forms and sample e-Evidence packages) and send them to their test gateway, in order to emulate the reception of such a message. In addition, the CTP can receive such messages and check them against the pre-approved XSDs to ensure the validity of the message’s structure. This can be very useful to get an idea of how e-CODEX messages are structured as well as quickly verify an application’s conformance when XSD schema changes are introduced.

3.2 Basic Functionality

3.2.1 Messages Module

This is a view of the messages related to the logged-in user account *only*. Messages are *grouped by conversationID* into collapsible panels with each panel row showing the basic information about that message, along with the latest status.



e-CODEX CTP Messages

- This page lists the messages that you send from the CTP to your gateway or vice-versa.
- Messages are grouped by Conversation ID.
- You can click on a Conversation ID to expand it and see the messages it is associated with

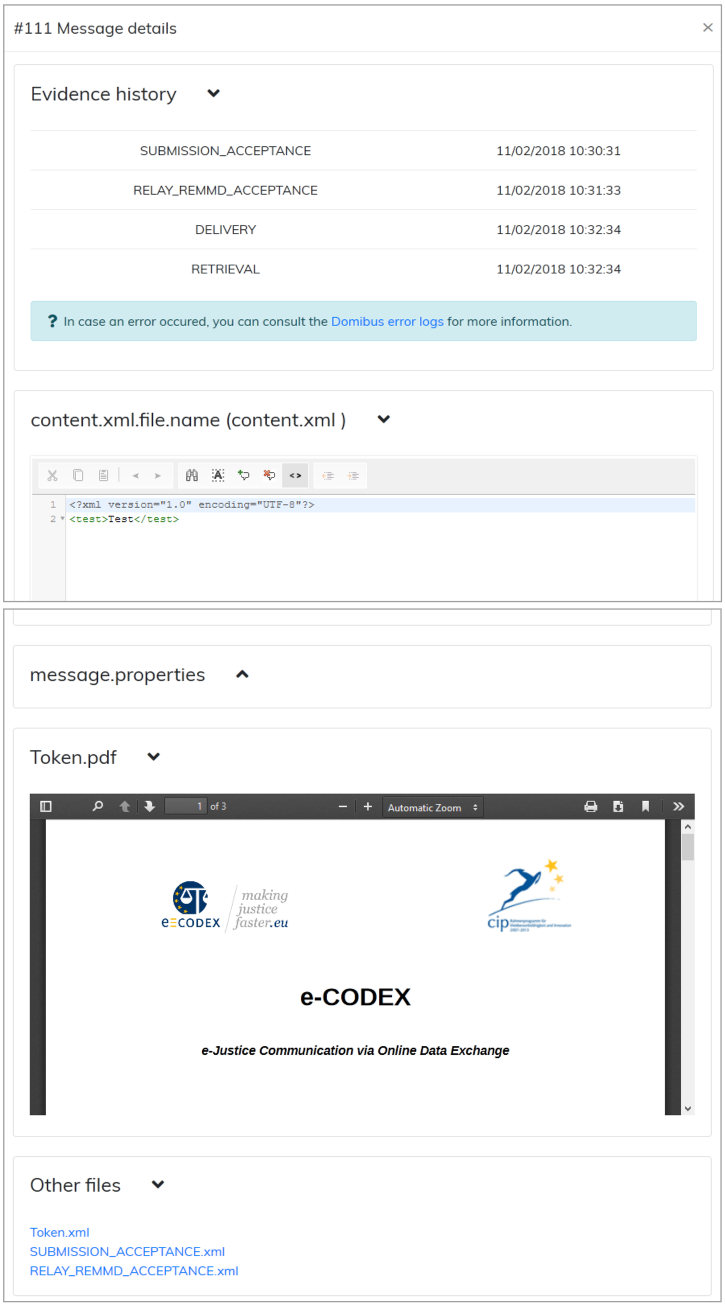
The messages below are part of this Conversation. Click on the ℹ icon next to each message for more information about it.

⇄	ID	From	To	Usecase	Action	Last evidence	Date created	Date modified	
↑	108	CTP	GR	GW-TEST	Test_Form	RETRIEVAL	05/02/2018 13:04:36	05/02/2018 13:06:47	ℹ

Figure 3: *The e-CODEX CTP Message Module (Outgoing Messages)*

The table is automatically refreshed every 10 seconds, so the delivery process can be easily monitored in-real time. The user has the ability to enable or disable this automatic update by clicking on the “Refresh” button (9).

A button entitled "details" (denoted by the information icon - ℹ) shows further information about each message, along with a full list of related evidences ("evidence history"), as they appeared on the e-CODEX CTP Gateway. The final versions of the content.xml, content.pdf and message.properties files of the message which was either sent or received are shown for reference, along with security tokens & attachments:



#111 Message details

Evidence history

SUBMISSION_ACCEPTANCE	11/02/2018 10:30:31
RELAY_REMMD_ACCEPTANCE	11/02/2018 10:31:33
DELIVERY	11/02/2018 10:32:34
RETRIEVAL	11/02/2018 10:32:34

? In case an error occured, you can consult the Domibus error logs for more information.

content.xml.file.name (content.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<test>Test</test>
```

message.properties

Token.pdf

e-CODEX

e-Justice Communication via Online Data Exchange

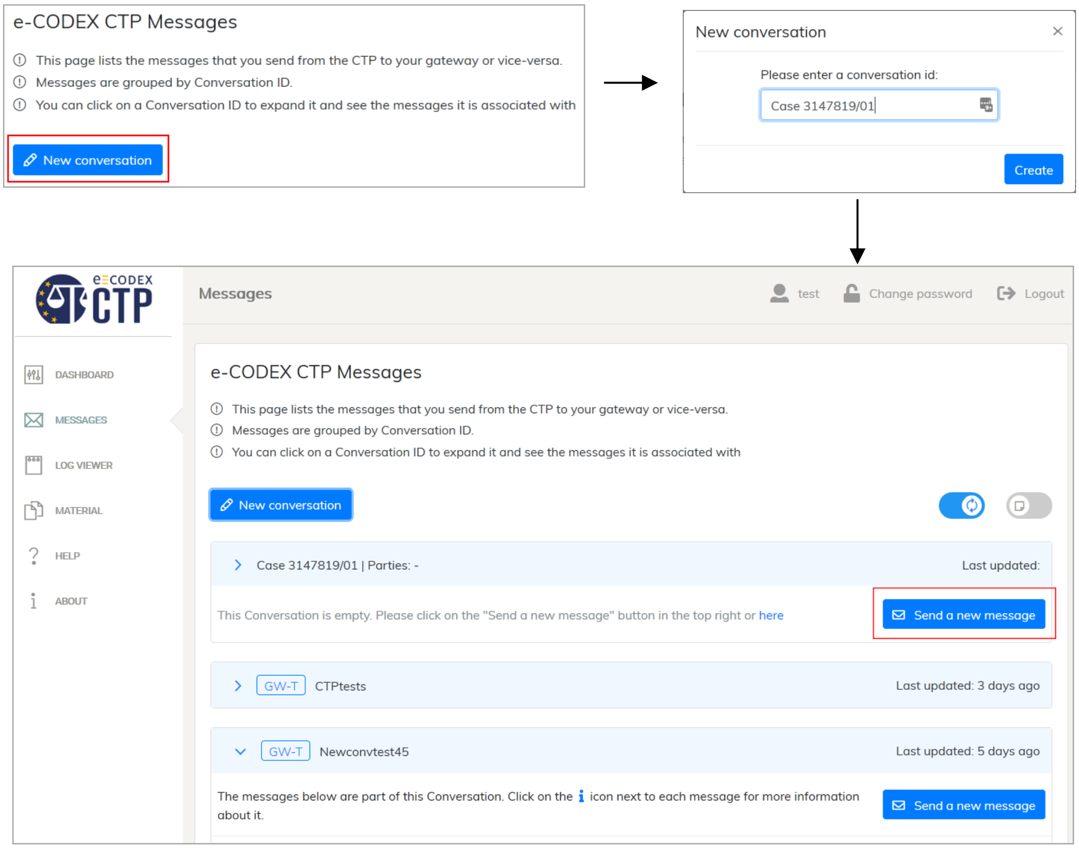
Other files

Token.xml
SUBMISSION_ACCEPTANCE.xml
RELAY_REMMD_ACCEPTANCE.xml

***Figure 4**: CTP: e-CODEX Message Details*

3.2.2 Sending a message to an e-CODEX gateway

At the core of the e-CODEX CTP is the "new outgoing message" functionality. To send a new message to their gateway, the user can either (1) choose an existing conversation ID and click on "Send a new message" or (2) create a new conversation ID and then click on the respective "Send a new message button":



e-CODEX CTP Messages

ⓘ This page lists the messages that you send from the CTP to your gateway or vice-versa.
ⓘ Messages are grouped by Conversation ID.
ⓘ You can click on a Conversation ID to expand it and see the messages it is associated with

New conversation

New conversation ×

Please enter a conversation id: Case 3147819/01

Create

Messages — test — Change password — Logout

DASHBOARD
MESSAGES
LOG VIEWER
MATERIAL
HELP
ABOUT

e-CODEX CTP Messages

ⓘ This page lists the messages that you send from the CTP to your gateway or vice-versa.
ⓘ Messages are grouped by Conversation ID.
ⓘ You can click on a Conversation ID to expand it and see the messages it is associated with

New conversation

Case 3147819/01 | Parties: - — Last updated:

This Conversation is empty. Please click on the "Send a new message" button in the top right or here — Send a new message

GW-T CTPtests — Last updated: 3 days ago

GW-T Newconvtest45 — Last updated: 5 days ago

The messages below are part of this Conversation. Click on the ℹ icon next to each message for more information about it. — Send a new message

__Figure 5__: CTP: New e-CODEX ConversationID

The related pop-up modal allows the user to define the characteristics of the message to be sent in two steps:

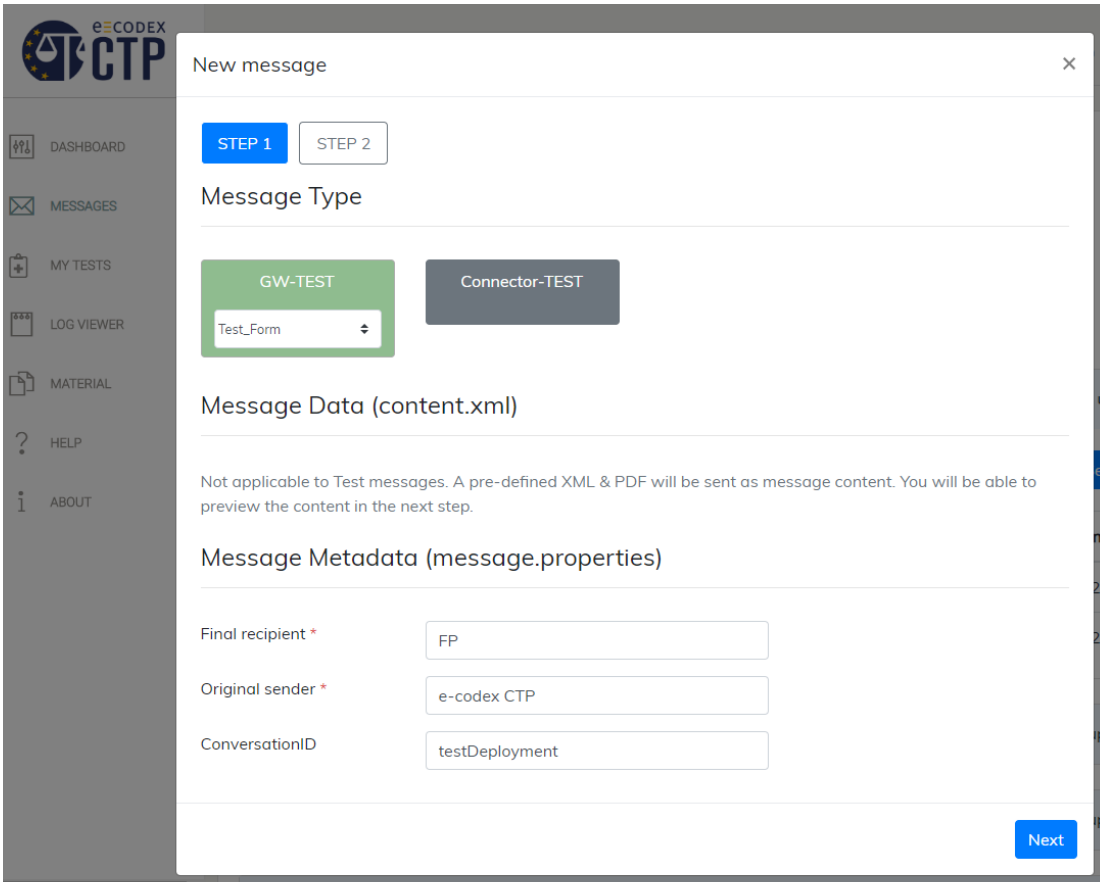
<u>STEP 1</u>

Message Type: Allows the user to select from the available Message Types.

Message Data: According to the message type selected, some or all of the following options are available:

- **SBDH data:** Content of the SBDH section of the e-CODEX message XML (Receiver/Sender identifier & authority)
- **Case ID:** Customise the content of the jus:IdentificationCaseIdentification/jus:NumberID XML Node (reply forms only)

Message Metadata: Contents of the message.properties file of the e-CODEX message, including the conversationID (auto-filled)



New message

Message Type

Message Data (content.xml)

Not applicable to Test messages. A pre-defined XML & PDF will be sent as message content. You will be able to preview the content in the next step.

Message Metadata (message.properties)

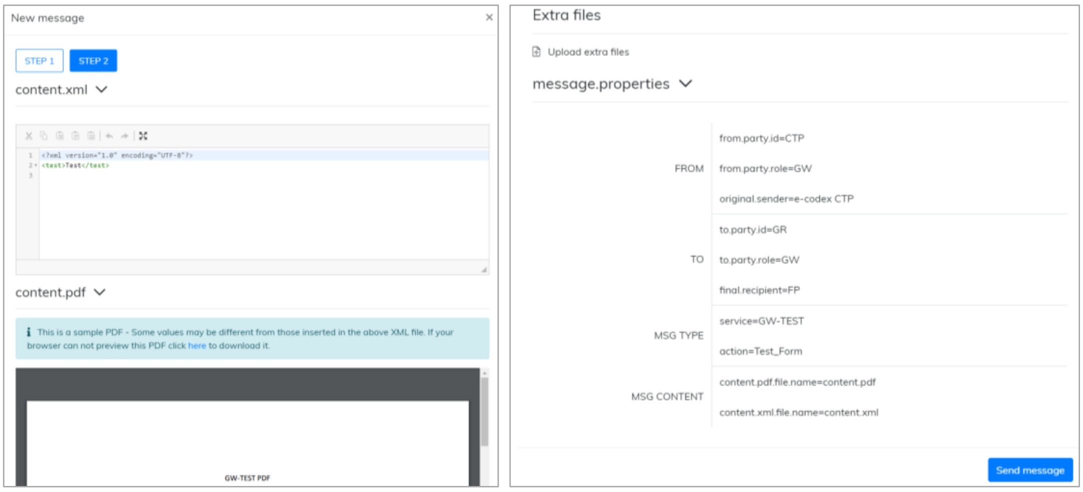
Final recipient *	FP
Original sender *	e-codex CTP
ConversationID	testDeployment

Figure 6: *CTP: New Outgoing Message – STEP 1*

STEP 2

In step 2, the user is able to preview the content of the outgoing message via a web text editor. This editor also offers colour-coding for XML files. The files that can be previewed are:

- **content**.xml
- **content**.pdf (available for downloading or embedded preview)
- **message**.properties (this file cannot be edited)



```
<?xml version="1.0" encoding="UTF-8"?>
<test>Test</test>
```

This is a sample PDF - Some values may be different from those inserted in the above XML file. If your browser can not preview this PDF click here to download it.

GW-TEST PDF

FROM	from.party.id=CTP
	from.party.role=GW
	original.sender=e-codex CTP

TO	to.party.id=GR
	to.party.role=GW
	final.recipient=FP

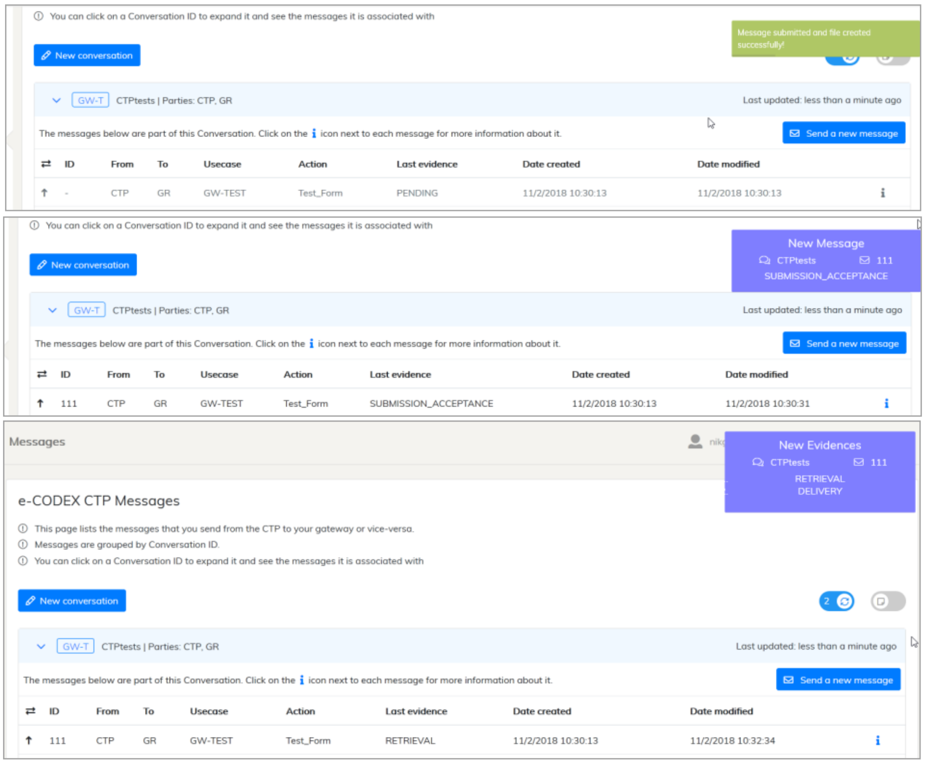
MSG TYPE	service=GW-TEST
	action=Test_Form

MSG CONTENT	content.pdf.file.name=content.pdf
	content.xml.file.name=content.xml

Figure 7: *CTP: New Outgoing Message – STEP 2*

3.2.3 Tracking message progress

All messages (either sent by the CTP or sent to the CTP) are tracked in real-time using a notification system. The “Last Evidence” section of each message row is also automatically updated:



You can click on a Conversation ID to expand it and see the messages it is associated with

Message submitted and file created successfully!

GW-T CTPtests | Parties: CTP, GR — Last updated: less than a minute ago

The messages below are part of this Conversation. Click on the i icon next to each message for more information about it.

⇄	ID	From	To	Usecase	Action	Last evidence	Date created	Date modified	
↑	-	CTP	GR	GW-TEST	Test_Form	PENDING	11/2/2018 10:30:13	11/2/2018 10:30:13	i

You can click on a Conversation ID to expand it and see the messages it is associated with

New Message
CTPtests 111
SUBMISSION_ACCEPTANCE

GW-T CTPtests | Parties: CTP, GR — Last updated: less than a minute ago

The messages below are part of this Conversation. Click on the i icon next to each message for more information about it.

⇄	ID	From	To	Usecase	Action	Last evidence	Date created	Date modified	
↑	111	CTP	GR	GW-TEST	Test_Form	SUBMISSION_ACCEPTANCE	11/2/2018 10:30:13	11/2/2018 10:30:31	i

New Evidences
CTPtests 111
RETRIEVAL
DELIVERY

e-CODEX CTP Messages

- This page lists the messages that you send from the CTP to your gateway or vice-versa.
- Messages are grouped by Conversation ID.
- You can click on a Conversation ID to expand it and see the messages it is associated with

GW-T CTPtests | Parties: CTP, GR — Last updated: less than a minute ago

The messages below are part of this Conversation. Click on the i icon next to each message for more information about it.

⇄	ID	From	To	Usecase	Action	Last evidence	Date created	Date modified	
↑	111	CTP	GR	GW-TEST	Test_Form	RETRIEVAL	11/2/2018 10:30:13	11/2/2018 10:32:34	i

Figure 8: *Message & Evidence notifications*

3.3 EXEC Testing tools plugin

For the purposes of the EXEC Project, a new plug-in has been developed for the CTP: EXEC Testing tools. This plugin is already implemented at https://exectest.ee.auth.gr and will facilitate testing in EXEC by providing web interfaces to assist both test administration and test reporting. This chapter will describe the basic functionality added by this plugin as well as provide a visual walkthrough through its functions and discuss its interconnection with the EXEC Testing XSD Schemas (presented in Chapter 2).

3.3.1 The CTP Test Administrator

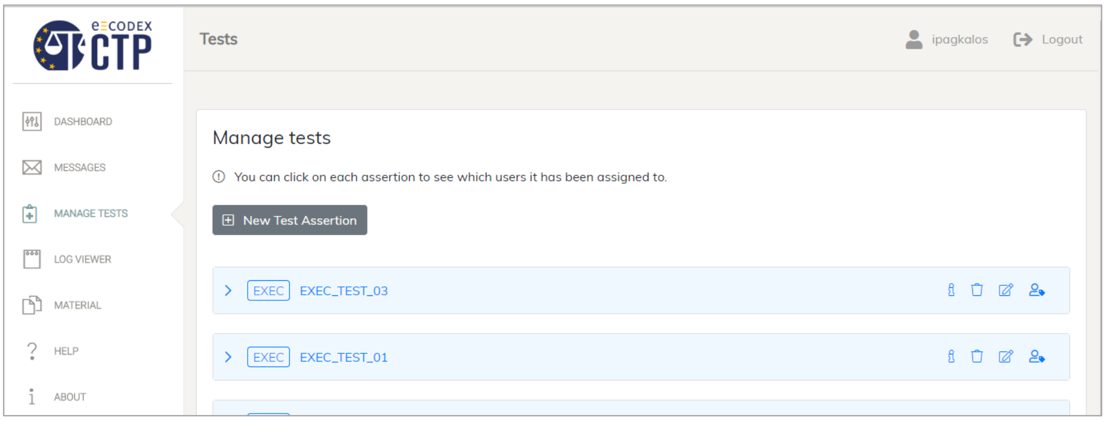
A new role on the CTP, entitled “Test Administrator” allows the respective users (e.g. WP4 leaders) to:

- create new Test Assertions
- assign test assertions to CTP users, according to the Test Plan agreed upon within the EXEC project
- monitor the progress of CTP users in regards to test completion and report generation

These functions are consolidated within the “Manage Tests” Web interface.

3.3.2 The “Manage Tests” Web interface

A new icon on the CTP sidebar, entitled “Manage Tests” will appear only to the CTP Test Administrator. Using this web interface, the test administrator is able to create new Test Assertions or assign existing ones to CTP users.

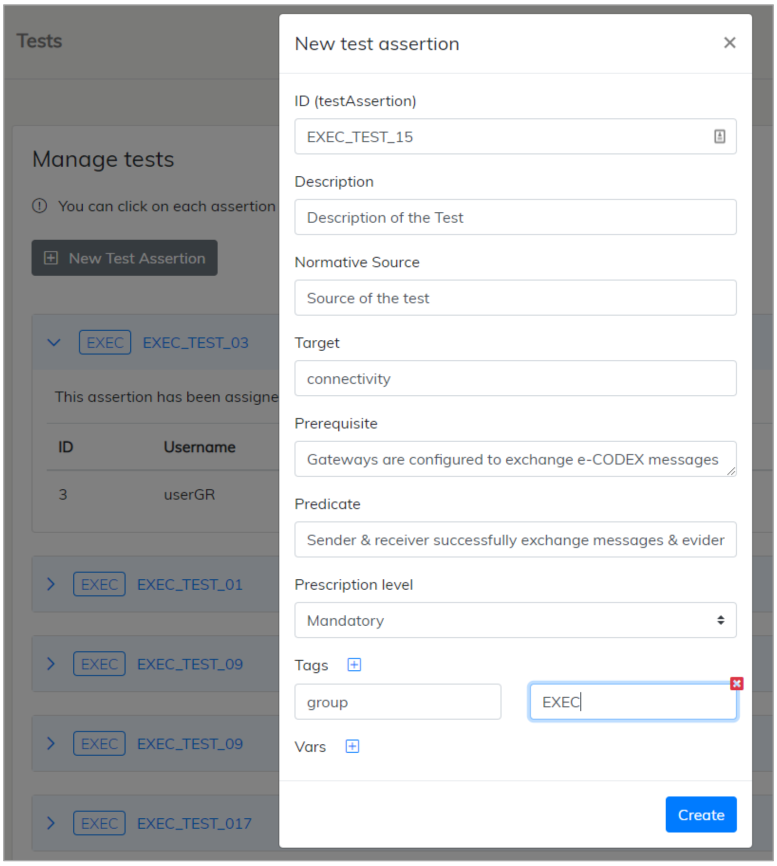


Manage tests

You can click on each assertion to see which users it has been assigned to.

Figure 9: *CTP: Manage Tests Interface*

The test administrator is able to create a new Test Assertion by clicking on the respective “New Test Assertion” button. A pop-up modal will guide the user through completing all respective elements of a test assertion, as discussed in Chapter 2.1. Including more detail such as the “group” tag, will allow the CTP to automatically group and “tag” tests by type and other variables.



New test assertion

ID (testAssertion): EXEC_TEST_15

Description: Description of the Test

Normative Source: Source of the test

Target: connectivity

Prerequisite: Gateways are configured to exchange e-CODEX messages

Predicate: Sender & receiver successfully exchange messages & evider

Prescription level: Mandatory

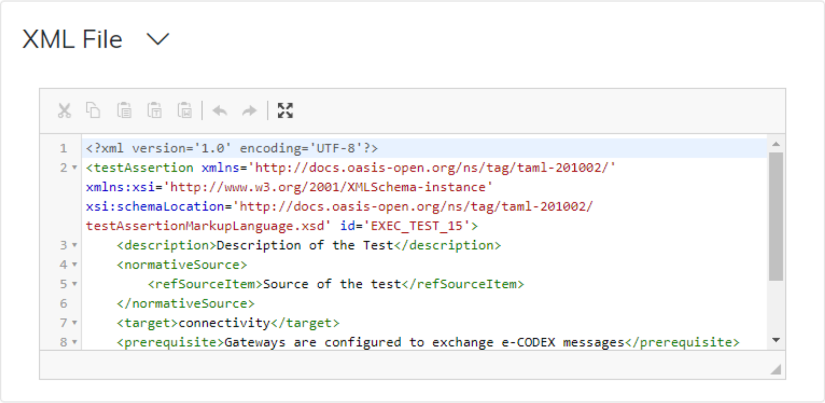
Tags: group | EXEC

Vars

Create

__Figure 10:__ CTP: Creating a new Test Assertion

After clicking on “Create”, the CTP will automatically create the respective XML file according to the EXEC Testing XSD schema (see Chapter 2.1), which provides a machine-understandable entity that can easily be shared and re-used.

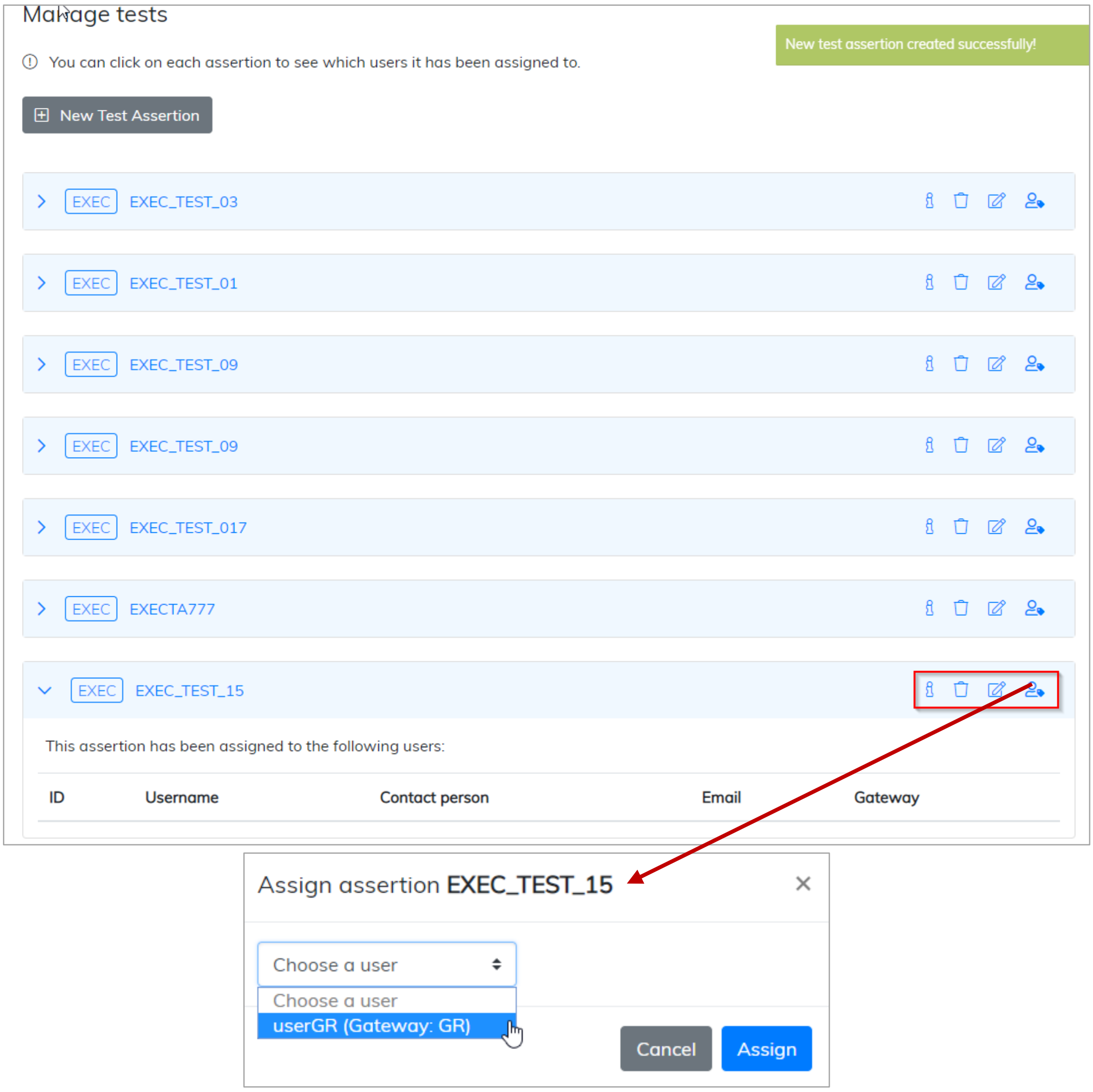


XML File

```xml
<?xml version='1.0' encoding='UTF-8'?>
<testAssertion xmlns='http://docs.oasis-open.org/ns/tag/taml-201002/'
xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:schemaLocation='http://docs.oasis-open.org/ns/tag/taml-201002/
testAssertionMarkupLanguage.xsd' id='EXEC_TEST_15'>
    <description>Description of the Test</description>
    <normativeSource>
        <refSourceItem>Source of the test</refSourceItem>
    </normativeSource>
    <target>connectivity</target>
    <prerequisite>Gateways are configured to exchange e-CODEX messages</prerequisite>
```

__Figure 11:__ CTP: Generating a new Test Assertion XML

Following the successful creation of a test, the test Administrator simply has to click on the respective icon in order to assign tests to CTP users:



Manage tests

New test assertion created successfully!

ⓘ You can click on each assertion to see which users it has been assigned to.

⊞ New Test Assertion

- › EXEC EXEC_TEST_03
- › EXEC EXEC_TEST_01
- › EXEC EXEC_TEST_09
- › EXEC EXEC_TEST_09
- › EXEC EXEC_TEST_017
- › EXEC EXECTA777
- ⌄ EXEC EXEC_TEST_15

This assertion has been assigned to the following users:

ID	Username	Contact person	Email	Gateway

Assign assertion **EXEC_TEST_15** ×

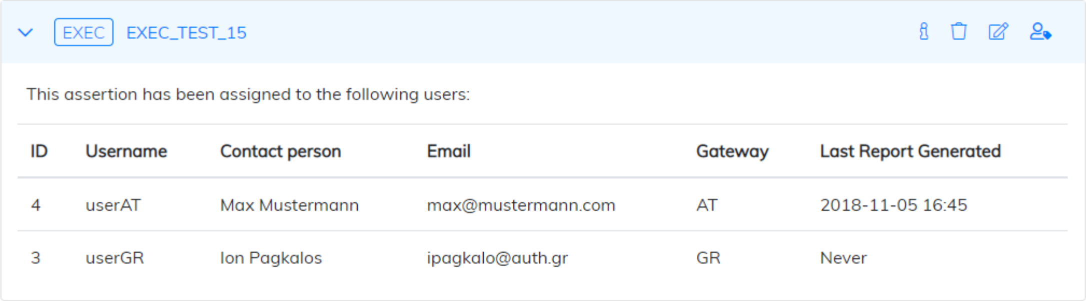
Choose a user

Choose a user
userGR (Gateway: GR)

Cancel Assign

Figure 12: *CTP: Assigning a Test Assertion to a User*

After successfully assigning the test to a user, the administrator is able to see from the same interface whether this user has submitted a test report on this test (the procedure for the user is described in the next section).



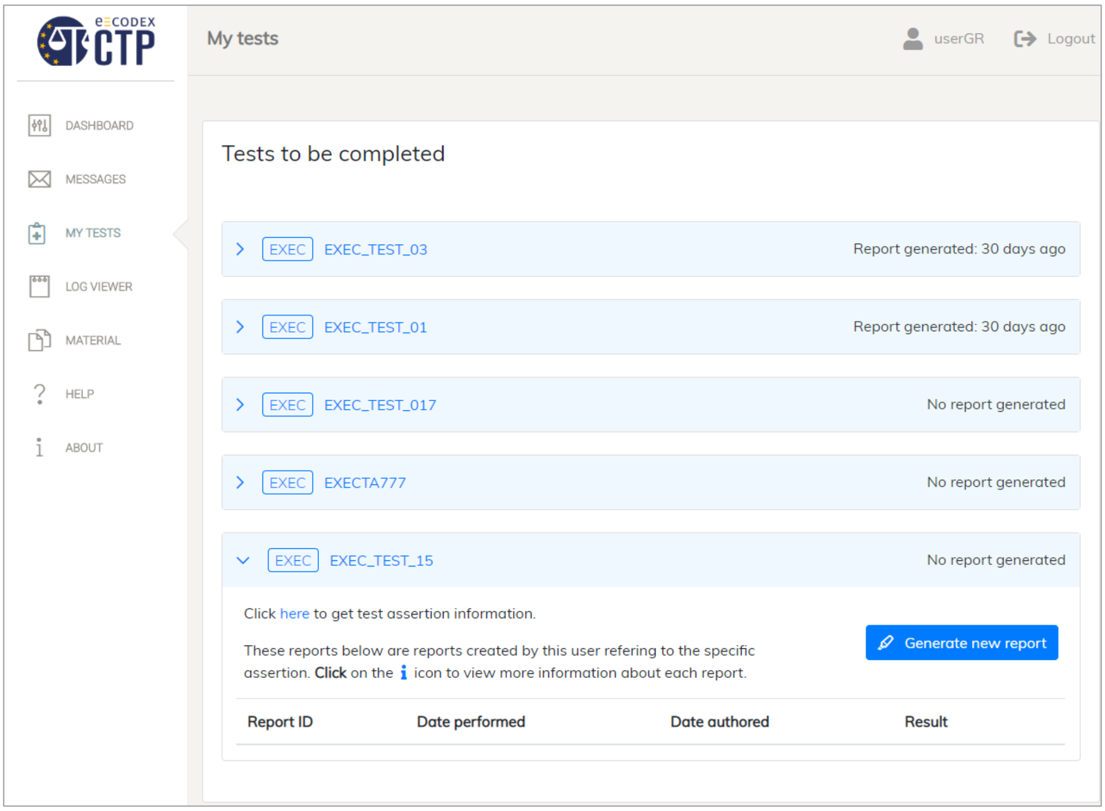
This assertion has been assigned to the following users:

ID	Username	Contact person	Email	Gateway	Last Report Generated
4	userAT	Max Mustermann	max@mustermann.com	AT	2018-11-05 16:45
3	userGR	Ion Pagkalos	ipagkalo@auth.gr	GR	Never

Figure 13: *Monitoring Test Assignment and Report generation*

3.3.3 The “My Tests” Web interface

Each registered user of the CTP now has access to the “My Tests” Web interface, accessible through the menu on the left. This interface will list all messages that are assigned to the user by the Tests administrator (see previous chapter) and allows the user to generate “Test Reports” for each test that has been completed



Tests to be completed

EXEC	EXEC_TEST_03	Report generated: 30 days ago
EXEC	EXEC_TEST_01	Report generated: 30 days ago
EXEC	EXEC_TEST_017	No report generated
EXEC	EXECTA777	No report generated
EXEC	EXEC_TEST_15	No report generated

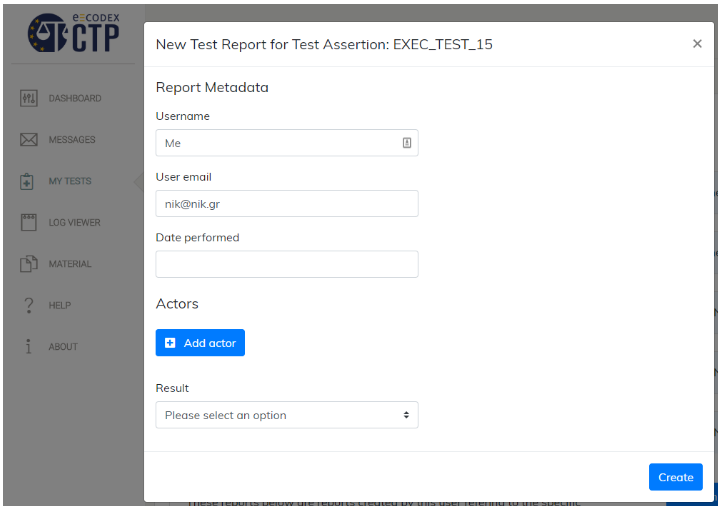
Click here to get test assertion information.

These reports below are reports created by this user refering to the specific assertion. **Click** on the ℹ icon to view more information about each report.

Report ID	Date performed	Date authored	Result

Figure 14: *The CTP My Tests Web interface*

In order to generate a report, the user can click on the respective button which will show a pop-up modal that guides the user through test reporting. As much information as possible is auto-filled in this report: for example, the username and email of the person reporting is copied from the user's CTP profile.



New Test Report for Test Assertion: EXEC_TEST_15

Report Metadata

Username: Me

User email: nik@nik.gr

Date performed:

Actors

Add actor

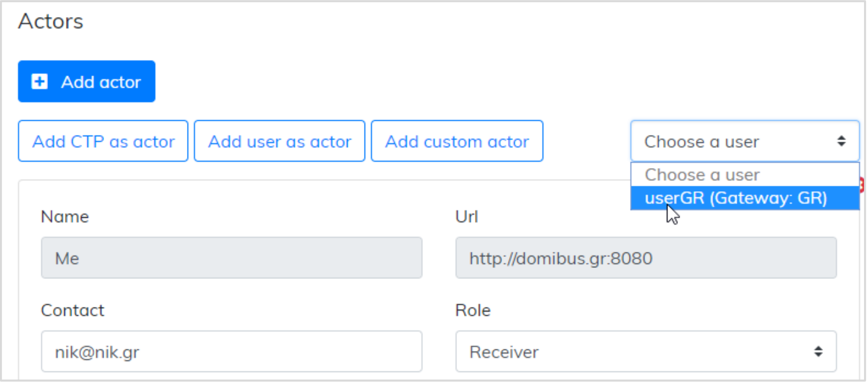
Result: Please select an option

Create

Figure 15: Creating a new Test Report

To add actors to each test, the new test report modal provides an intuitive interface that allows users to select:

- Adding the CTP as actor: This will auto-fill the CTP's details and allow the user to define the CTP as either the receiver or the sender of the test message
- Adding a CTP user as actor: This will auto-fill the user's details (including the server URL, as defined in the CTP user profile) and also allows selecting the user as sender or receiver of the test message
- Adding a "custom" actor: This allows the user to enter all information regarding the actor manually



Actors

Choose a user: Choose a user / userGR (Gateway: GR)

Name: Me

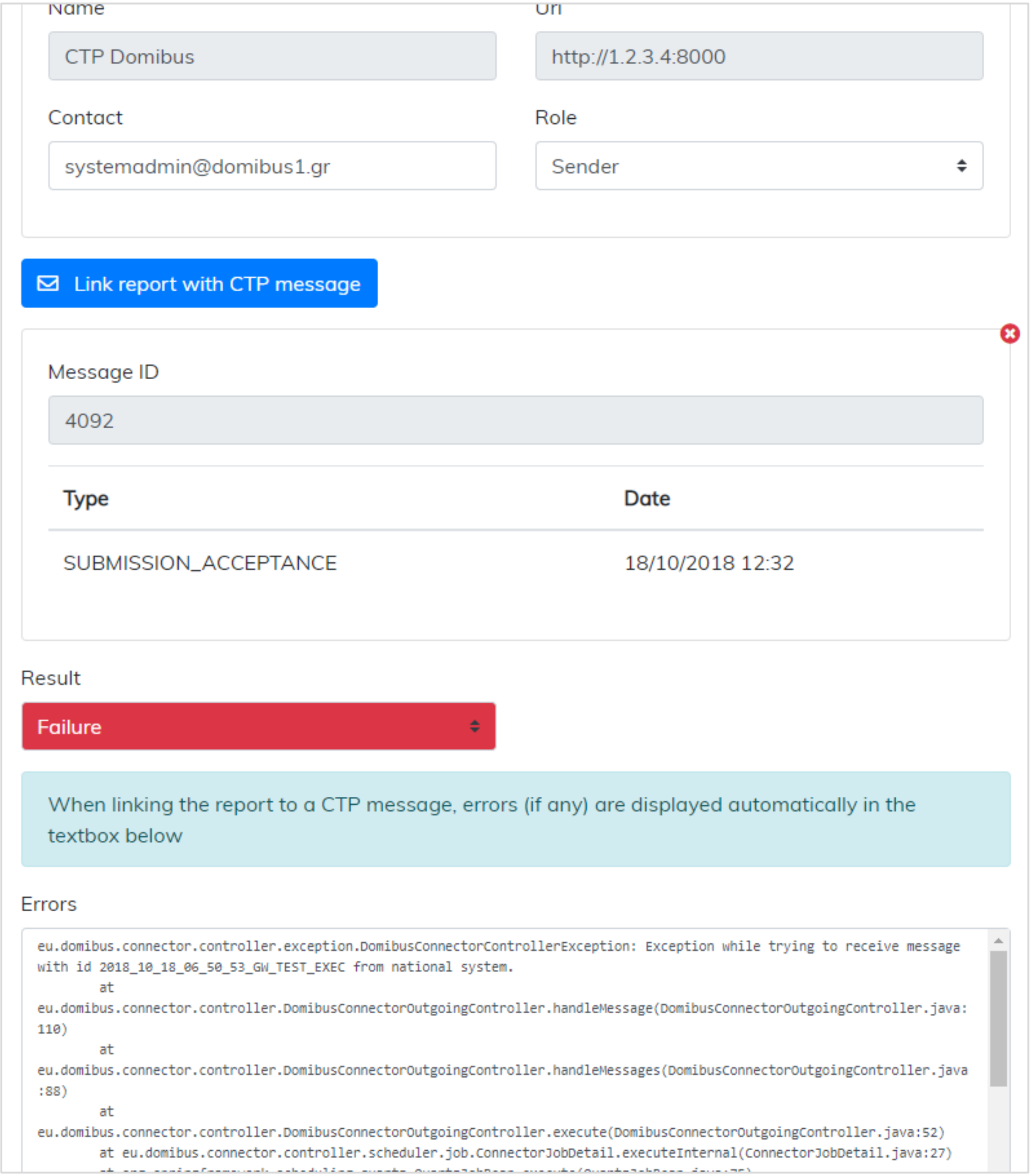
Url: http://domibus.gr:8080

Contact: nik@nik.gr

Role: Receiver

***Figure 16:** Adding a new Actor on a CTP Test Report*

When the CTP is part of a test report, a new function becomes available: Linking a Report with a CTP Message. This allows the users to *automatically retrieve evidences as well as error logs from the CTP's connector database*, in order to more accurately track and report test progress:



Name	Url
CTP Domibus	http://1.2.3.4:8000
Contact	Role
systemadmin@domibus1.gr	Sender

Message ID

4092

Type	Date
SUBMISSION_ACCEPTANCE	18/10/2018 12:32

Result

Failure

When linking the report to a CTP message, errors (if any) are displayed automatically in the textbox below

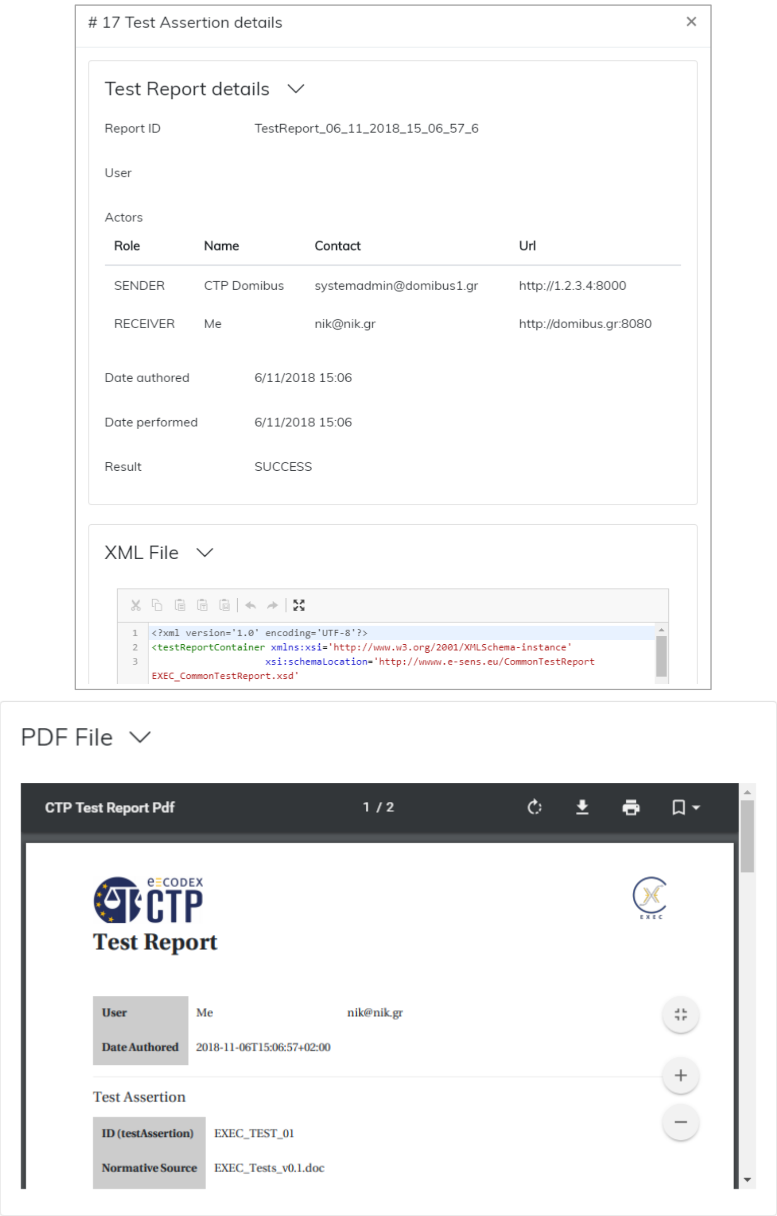
Errors

```
eu.domibus.connector.controller.exception.DomibusConnectorControllerException: Exception while trying to receive message
with id 2018_10_18_06_50_53_GW_TEST_EXEC from national system.
        at
eu.domibus.connector.controller.DomibusConnectorOutgoingController.handleMessage(DomibusConnectorOutgoingController.java:
110)
        at
eu.domibus.connector.controller.DomibusConnectorOutgoingController.handleMessages(DomibusConnectorOutgoingController.java
:88)
        at
eu.domibus.connector.controller.DomibusConnectorOutgoingController.execute(DomibusConnectorOutgoingController.java:52)
        at eu.domibus.connector.controller.scheduler.job.ConnectorJobDetail.executeInternal(ConnectorJobDetail.java:27)
```

Figure 17: *Linking a Test report with a CTP message*

Finally, clicking on Generate report will

- Generate an XML report based on the EXEC Test Reports Schema
- Generate a PDF proof (a basic visual representation of XML data)
- Store the report in the CTP's database, making it available to Test administrators



17 Test Assertion details

Test Report details

Report ID	TestReport_06_11_2018_15_06_57_6
User	

Actors

Role	Name	Contact	Url
SENDER	CTP Domibus	systemadmin@domibus1.gr	http://1.2.3.4:8000
RECEIVER	Me	nik@nik.gr	http://domibus.gr:8080

Date authored	6/11/2018 15:06
Date performed	6/11/2018 15:06
Result	SUCCESS

XML File

```
<?xml version='1.0' encoding='UTF-8'?>
<testReportContainer xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
                     xsi:schemaLocation='http://wwww.e-sens.eu/CommonTestReport
EXEC_CommonTestReport.xsd'
```

PDF File

CTP Test Report Pdf 1 / 2

Test Report

User	Me	nik@nik.gr
Date Authored	2018-11-06T15:06:57+02:00	

Test Assertion

ID (testAssertion)	EXEC_TEST_01
Normative Source	EXEC_Tests_v0.1.doc

Figure 18: *Generating a new XML & PDF Test report*

4 EXEC Test Strategy

This section provides a high-level overview of the Testing Strategy that will be followed by member states participating in the project. Taking into account past experience in e-CODEX testing, EXEC partners will follow a high-level testing strategy that is comprised of a set of initial automated tests with the CTP, followed by End-2-End exchanges between member states. All testing will be monitored and coordinated via the CTP.

4.1 Basic Testing Strategy

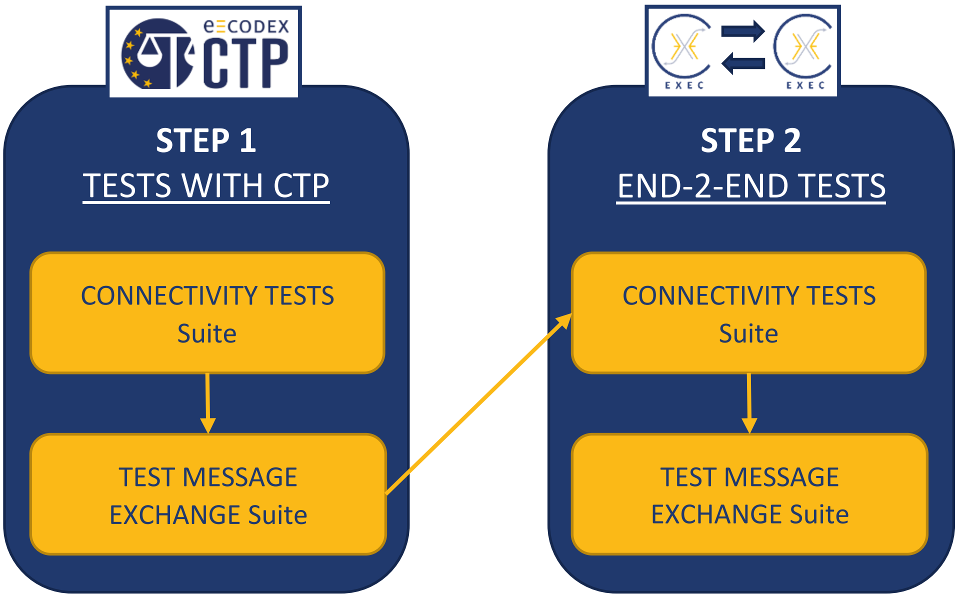


Figure 19: *EXEC Testing Strategy (high-level)*

As can be seen in **Figure 19**, the basic Testing strategy of EXEC is a two-step approach:

Step 1 involves the exchange of test messages between (i) member states and (ii) the e-CODEX Central Testing Platform.

Step 2 involves the End-2-End exchange of test messages between member states, using the EXEC Testing environment (independent of production systems).

In order to allow countries to move forward in *waves* and be able to use experience sharing, End-2-End testing (step 2) will be assigned by EXEC WP4 to *groups* of member states, based on factors like:

- Readiness
- Testing groups of alike implementations
- Testing in groups on geographic proximity

In addition, EXEC will also test a sample of ‘cross contexts’, i.e. a combination of the factors mentioned above.

4.2 Basic Testing Suites

Both steps of the basic testing strategy include the same set of two test suites, depending on the *content* of the test messages being exchanged:

4.2.1 Connectivity Tests

Connectivity Tests involve the exchange of simple, non-content-specific e-CODEX messages, mainly using the GW-TEST & CON-TEST services. This test suite can include, for example, tests based on the functional requirements as specified by the EC’s e-Evidence guidelines, the specific capabilities and nuances of EXEC’s infrastructure (Reference Implementation, Connector, Gateway) as well as non-functional requirements, such as load handling/balancing, availability/redundancy and access control. An example of basic steps in a message exchange between members in such a scenario can be seen in **Table 2** & **Table 3** (based on previous e-CODEX tests)

Test steps	Description test	Expected result and checks
Step 1	Gateway in Country A sends a message to Gateway in Country B	• The message files are in the corresponding out-directories of the backend interface A • DB entries for the sent messages in the DB • Corresponding log entries in the DB
Step 2	Gateway in Country B receives a message	• The message files are in the corresponding in-directories of the backend interface B • DB entries for the received messages in the DB • Corresponding log entries in the DB
Step 3	Gateway in Country B sends a message to Gateway in Country A	• The message files are in the corresponding out-directories of the backend interface B • DB entries for the sent messages in the DB • Corresponding log entries in the DB
Step 4	Gateway in Country A receives a message	• The message files are in the corresponding in-directories of the backend interface A • DB entries for the received messages in the DB • Corresponding log entries in the DB

Table 2: *Basic steps of a GW-TEST message exchange*

Test steps	Description test	Expected result and checks
Step 1	Sending a test document with national application A to connector A	• The message stored in Connector DB • Hash stored • Log Entry available • No Exceptions
Step 2	Validation of signature	Trust-OK Token xml and PDF are created accordingly and containing the correct information
Step 3	Generation of ASiC-S Container	ASiC-S Container is available and includes expected documents (Form PDF and Token PDF)
Step 4	Forwarding Message to the GW A	• Periodic Timer Jobs are running • Log Entry in GW A available • No Exception occurs on Backend Interface and GW
Step 5	Sending Message to the GW B	• The message files are in the corresponding in-directories of the backend interface B • DB entries for the received messages in the GW DB • Corresponding log entries in the GW DB
Step 6	Forwarding Message to Connector B	• Evidence ○ created with national Message Id ○ Hash and Signed ○ Evidence received at National Application with correct national Message ID

Table 3: *Basic steps of a CON-TEST message exchange*

4.2.2 Test Message exchanges

Test Message exchanges involve the exchange of test (sample) EIO Forms, as well as the combination of EIO Forms + E-Evidence Package, as defined by the results of the collaboration between the EC, EXEC and the EVIDENCE2e-CODEX project. Detailed description of this tests will be provided in D4.2, but the high-level goal is to make sure that at least *one test for each EIO form type* is covered within tests. This includes:

- ANNEX A - EUROPEAN INVESTIGATION ORDER (EIO)
- ANNEX B - CONFIRMATION OF THE RECEIPT OF AN EIO
- ANNEX C - NOTIFICATION

Following the results of EXEC WP3, and more specifically, the successful installation of EXEC infrastructure by all partners, these tests will be modified according to two groups of users: (i) those using the EC's Reference Implementation for e-Evidence and (ii) those using their national solution. Even though it is not yet clear how different each solution will be, it is, however, not expected that tests will differ on a basic level, as all solutions should be compliant to the EU regulation and e-

CODEX requirements. It should also be noted that the goal within EXEC is not to test each solution's processes individually, but the overall successful technical exchange of forms on an End-2-End level.

4.3 Test Reporting, Progress monitoring and follow-ups

As discussed in chapter 3.3.3 ("My Tests" Web interface), EXEC members are able to use the CTP to track tests that have to be performed, their due date, as well as report on these tests when they are complete.

The tests administrator (WP4) is also able, using the same interfaces, to collect these reports for further analysis. This analysis can be performed "by hand" or by a software that will parse and process the XML reports (to be determined at a later stage).

However, specific issues that arise during testing will be expressed in these XML reports, catalogued on the CTP and resolved with the help of the appropriate actor:

- CEF Gateway: DIGIT
- e-Evidence RI: DGJUST/Subcontractor
- e-CODEX Connector: Me-CODEX consortium

In order to facilitate troubleshooting, the CTP will, over time, also include "known solutions to known problems".

Conclusion

This document described the preparatory work that took place in order to move forward with testing in EXEC. The result of this work is referred to as the EXEC Testing Toolset, comprised of (i) the EXEC Test Assertion & Test Reporting schemas and (ii) the specifically designed test reporting and monitoring plugin in the e-CODEX Central Testing Platform (CTP). In addition, this document also described the high-level strategy for testing that will take advantage of the new testing tools.

This more structured approach, coupled with the appropriate Web interfaces in a central environment (the CTP) is expected to help reduce the effort in documenting, sharing and – subsequently – collecting and analysing tests. The combination of XML-based reporting on the CTP and the web interface upgrades to handle them, also pave the way for its transition from an e-CODEX specific tool to a more generic e-Delivery testing solution, which is not only useful for the EXEC project, but may also be reused in other e-Justice contexts.

Following this deliverable, WP4 will create, share and schedule tests in collaboration with EXEC partners. These tests will be described in detail in Deliverable 4.2.