2. Developer Portal Overview

2.1 Pre-requisites

The Developer Portal itself is a web-based application and can be run from any modern browser such as Google Chrome, Microsoft Edge or Apple Safari.

The SDK is a Java based JAR file that can run on all leading platforms including Windows, Linux and Mac. The Java SDK (JAR) will run on JDK versions >=11 and <15, to comply with secp256k1 as per ZATCA security regulations.

The Integration Sandbox APIs can be accessed from all leading platforms as those mentioned above. REST APIs can be accessed from any Rest Client tools (Postman) for testing or using any coding languages (java, .Net, PHP, Nodejs, etc.) to call the rest services using HTTPs Protocol.

2.2 Structure / Sitemap

The Developer Portal is comprised of the following:

<table>
<tr><th colspan="4">Developer Portal</th></tr>
<tr><td colspan="2">Login</td><td>Access Portal Based Validator</td><td>Access Developer Portal Support Page</td></tr>
<tr><td>Access SDK Page</td><td>Access Integration Sandbox Page</td><td></td><td></td></tr>
<tr><td>• Download SDK
• SDK Support
• SDK Documentation
• SDK Version History</td><td>• Access API Documentation (Swagger Files)
• Test APIs for Onboarding, Renewal, Reporting and Clearance</td><td>Validate XMLs</td><td>Access FAQs</td></tr>
<tr><td colspan="4">Outside Developer Portal</td></tr>
<tr><td>Using the SDK</td><td colspan="2">Using the Integration Sandbox (APIs)</td><td></td></tr>
<tr><td>• Test compliance of XML
• Test compliance of QR Code (Generation Phase)
• Test Compliance of QR Code (Integration Phase)</td><td colspan="2">• Test APIs to obtain Compliance CSID and Production CSID (as part of Onboarding process)
• Test APIs to obtain new Compliance CSID and Production CSID (Test the Renewal process)
• Test API to submit documents for Reporting
• Test API to submit documents for Clearance</td><td></td></tr>
</table>

2.3 User Journeys

The recommended steps for Solution Developers are:

1. Read the XML Implementation Standards, Security Features Implementation Standards and Data Dictionary
2. Access the Developer Portal
3. Create a Developer Portal Account
4. Login to the Developer Portal as a Registered User
5. Access the SDK Page
6. Read the SDK Support and Documentation
7. Download the SDK
8. Test XML compliance using the SDK via CLI / local integration
9. Access the Integration Sandbox Page
10. Go through the API Documentation on Swagger
11. Test the APIs through Swagger
12. Test the APIs via integration
13. Leveraging the Developer Portal Support page FAQs for troubleshooting

The recommended steps for Non-technical users are:

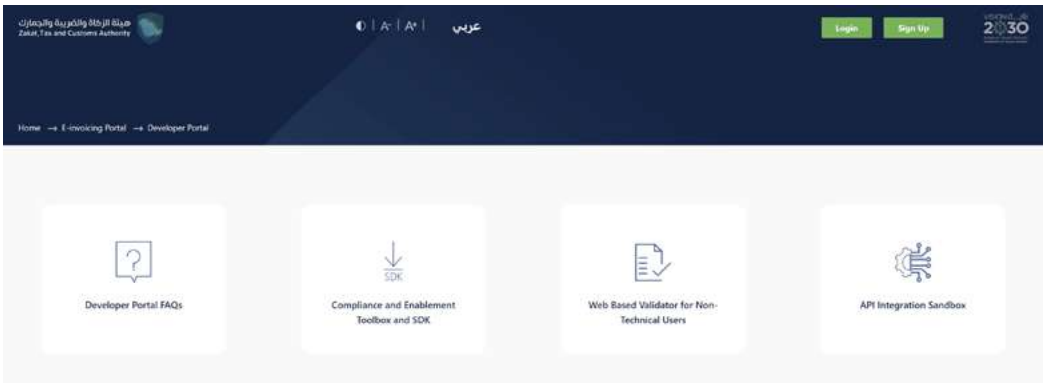
1. Access the Developer Portal
2. Accessing the Compliance and Enablement Toolbox Portal Page
3. Test XML compliance
4. Provide the error messages / responses (if any) to Solution Developers
5. Leveraging the Developer Portal Support page FAQs for troubleshooting

2.3.1 Accessing the Developer Portal

The process for accessing the Developer Portal is as follows:

1. Access the Developer Portal through the following weblink (https://sandbox.zatca.gov.sa/).
2. The user is directed to the Developer Portal main dashboard / landing page
    1. In this page the user can access the below sections without registration or login:
        1. Developer Portal Support Page which includes the FAQs.
        2. Web Based Validator for Non-Technical Users.
    2. The following sections would require the user to create a Developer Portal account:
        1. Compliance and Enablement Toolbox SDK Page.
        2. Integration Sandbox Page.

Note: The User can chose to toggle the language between English and Arabic by using the icon on the top right-hand side of the page.



Developer Portal main landing page

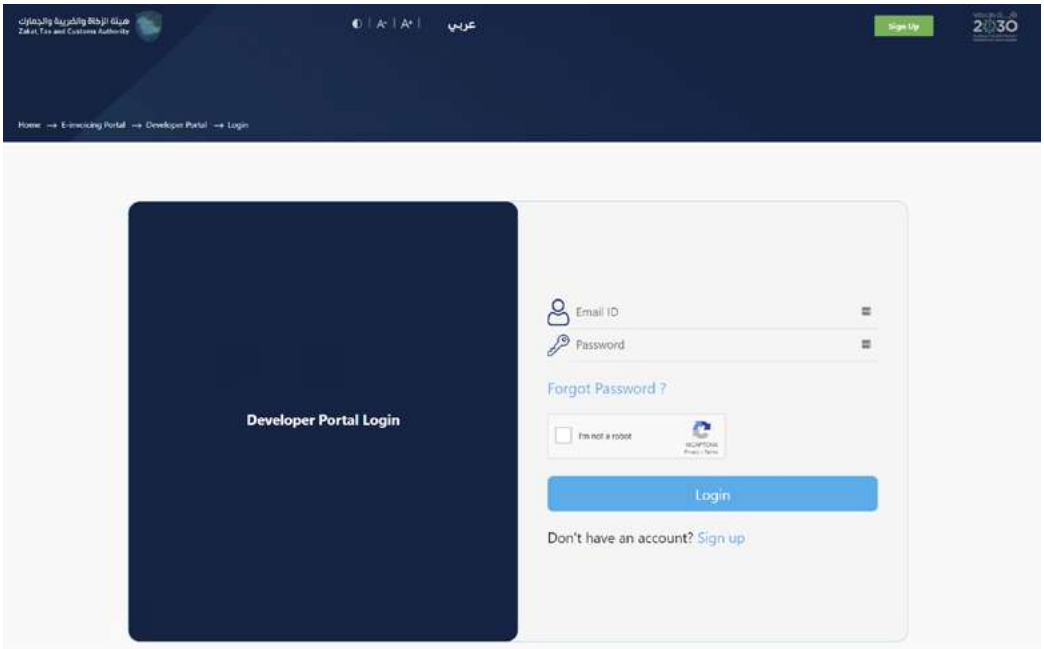
2.3.2 Creating a Developer Portal Account

As mentioned above, a Developer Portal account is required for accessing the Compliance and Enablement Toolbox SDK page and the Integration Sandbox page. You can ignore this step if you only wish to access the Web Based Validator or the Developer Portal Support page.

Once the user is on the main dashboard of the Developer Portal, they can click on the "Sign up" button at the top right-hand side as seen in the Figure below.

- Email ID
- First Name
- Last Name
- Company Name (optional field)
- Password
- Confirm Password

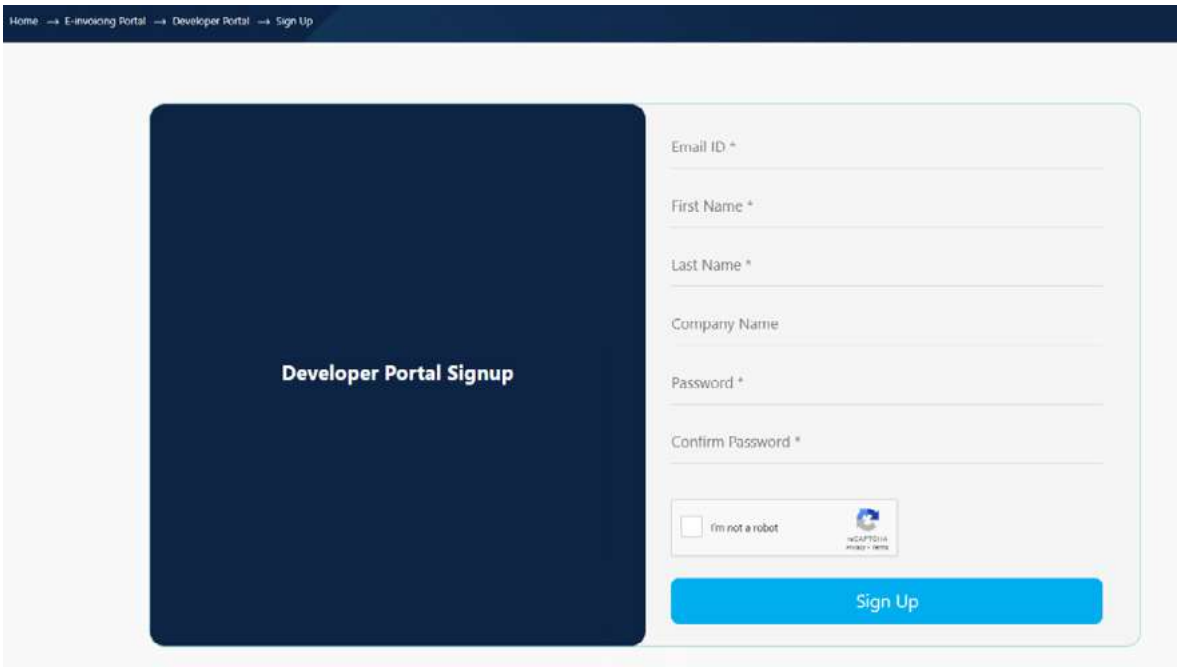
In the Sign Up page (as seen in the Figure below), the user will be prompted to create a new account by providing the following details:
The email must be a valid email and the password must be at least 8 characters comprising of at least one number, one letter each in lower and upper case, and one symbol.
After completing all the necessary fields, the user should click on the CAPTCHA verification followed by the Sign up button.



Login Page

After the user has signed up and created their account credentials, they can proceed to the Login page where they will be prompted to:

- Fill in the User Name and Password (as created by the user).
- Click the CAPTCHA.
- In addition, the user can click "Forgot Password"
- In the case where the user does not have an account set up and requires one, the user can click on the Sign Up option, in order to create a new account and proceed to the process described in this Section 2.3.2 of the User Manual for registration.
- After filling in all the information, the user should click on the Login button in order to proceed to the main dashboard again where the user will now also be able to access the Compliance and Enablement Toolbox SDK page and the Integration Sandbox page.
- A logged in user can logout at any time by clicking on the logout option on the header. The user can also change the password at any time by clicking on the arrow next to the user profile icon in header.

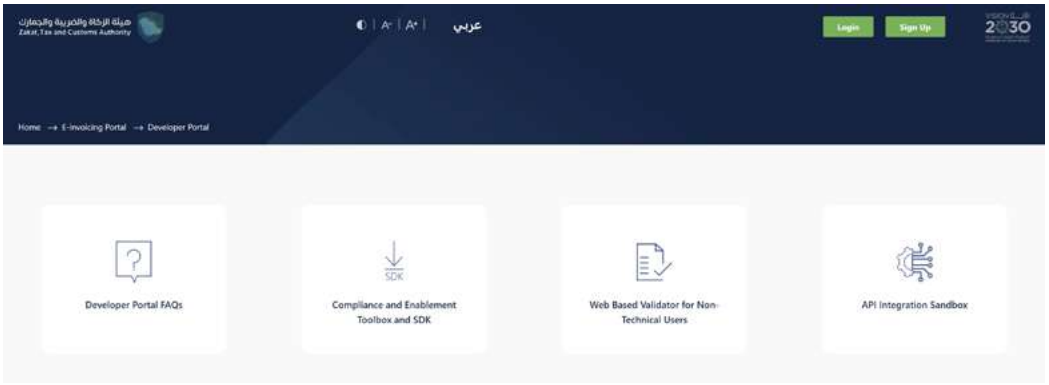


2.3.3 Accessing the Compliance and Enablement Toolbox SDK Page

The Compliance and Enablement Toolbox (SDK) which is an offline downloadable tool that can be used to validate an XML based e-invoice, credit or debit note files in accordance with the ZATCA published XML Implementation Standards. It also allows validation of the QR codes as per the prescribed structure. Developers can integrate their EGS units with the SDK locally (offline) or also test using a CLI.

The process for accessing and downloading the Compliance and Enablement Toolbox SDK through the Developer Portal is as follows:

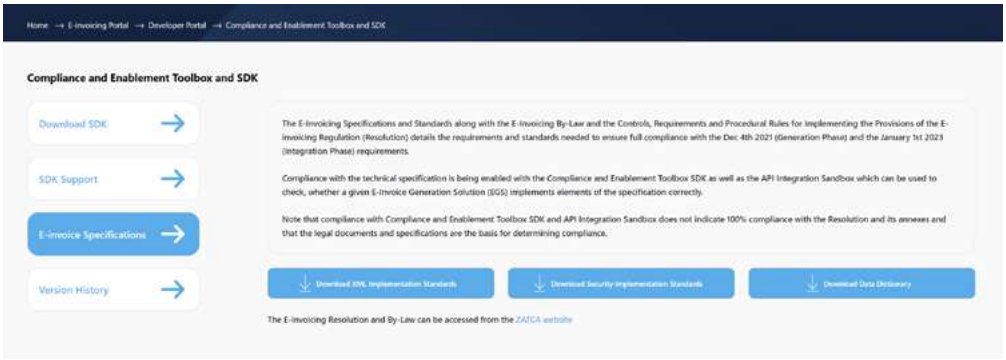
- The user should be registered and logged into the Developer Portal successfully
- The user should click on "Compliance and Enablement Toolbox and SDK" to view the SDK functionalities.



Accessing the Compliance and Enablement Toolbox (SDK)

After the user has accessed the Compliance and Enablement Toolbox SDK Page, the user can:

- Access the SDK support, which includes aspects such as how to use the SDK and how it works, as well as the minimum software requirements and the instructions of relevance to each Operating System/ environment.
- Access documentation such as the XML Implementation Standards (E-Invoice XML Implementation Standard), Security Features and Implementation Standards (E-Invoice Security Features and Implementation Standards) & Data Dictionary (E-Invoice Data Dictionary)
- Download the SDK after accepting the terms and conditions.
- View the version history which contains earlier releases of the SDK.

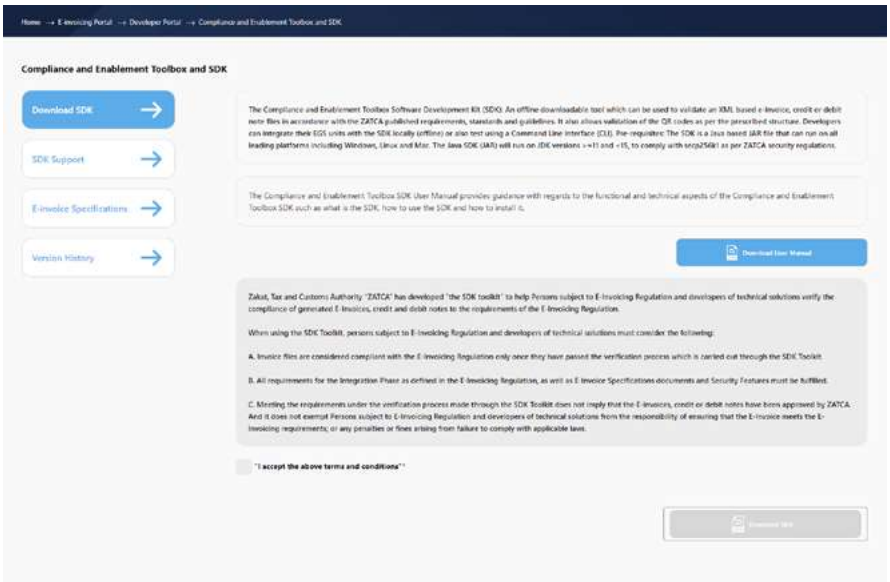


Accessing the E-Invoicing specification documents

2.3.4 Downloading the SDK

In order to download the SDK, the process is as follows:

- The user clicks on "Download SDK"
- The user has to click on "I accept the above terms and conditions"
- As the above is clicked, the "Download SDK" button will be activated and become available for the user to click on



downloading the SDK

2.3.5 Using the SDK (outside of the Developer Portal)

Please refer to the ZATCA E-Invoice Java SDK (CLI) Manual on the below link by downloading the SDK and then navigate to readme folder.

https://zatca.gov.sa/ar/E-Invoicing/SystemsDevelopers/ComplianceEnablementToolbox/Pages/DownloadSDK.aspx.

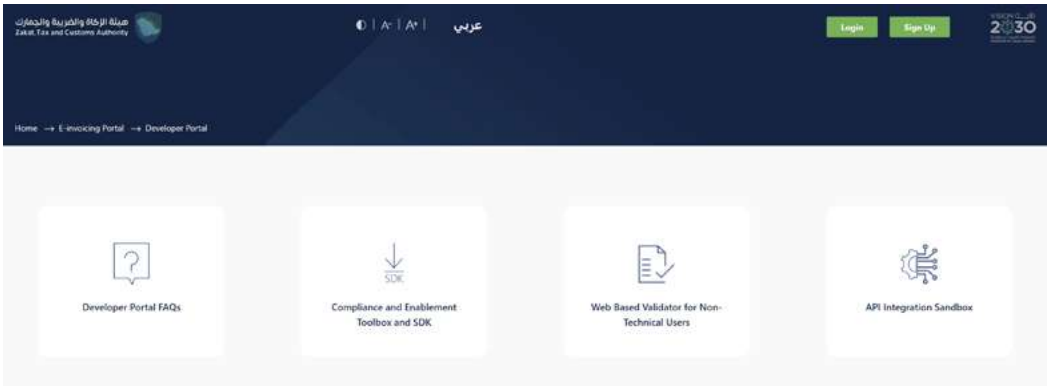
2.3.6 Accessing the Web Based Validator for Non-Technical Users

The user can test - using a web portal - the compliance of the XMLs of standard e-invoices, credit or debit notes generated so that they can know if they are in line with the ZATCA e-invoicing specifications and regulations or so that they can be alerted to any errors which are causing non-compliance with the ZATCA specifications and regulations. It is aimed at intermediate or non-technical users to validate XML based e-invoices, credit or debit note files from the portal directly, i.e. without the need to download the SDK or possess the technical know-how to run it.

This section details the process of accessing the "Web Based Validator for Non-Technical Users" in order to test the compliance of the e-invoice, credit and debit note XMLs. Users can access the "Web Based Validator for Non-Technical Users" Page through the Developer Portal (no prior registration or login is required). On this page, users can view information related to what the Web Based Validator aims to achieve and the user can access this and begin uploading the XMLs that they would want to test and validate.

The process for accessing the "Web Based Validator for Non-Technical Users" page is as follows:

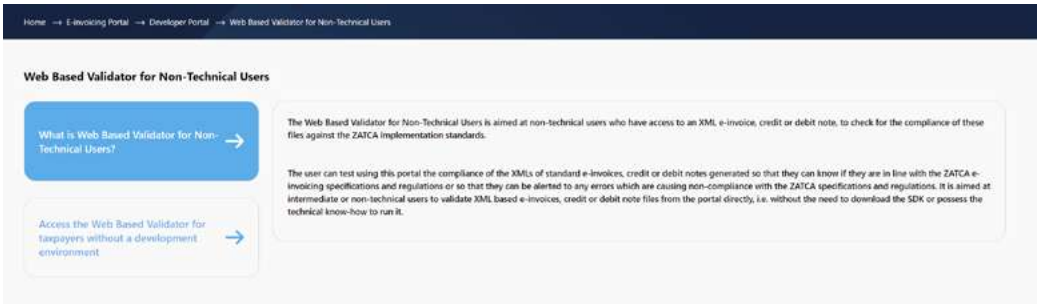
- The User accesses the "Web Based Validator for Non-Technical Users" on the Developer Portal (no prior registration or log in required).



Accessing the web based validator

- On the "Web Based Validator for Non-Technical Users", users can view information related to an explanation of the Web Based Validator and what it aims to do

- In addition, users can click on "Access the Web Based Validator for taxpayers without a development environment" in order to begin testing and validating their XMLs.

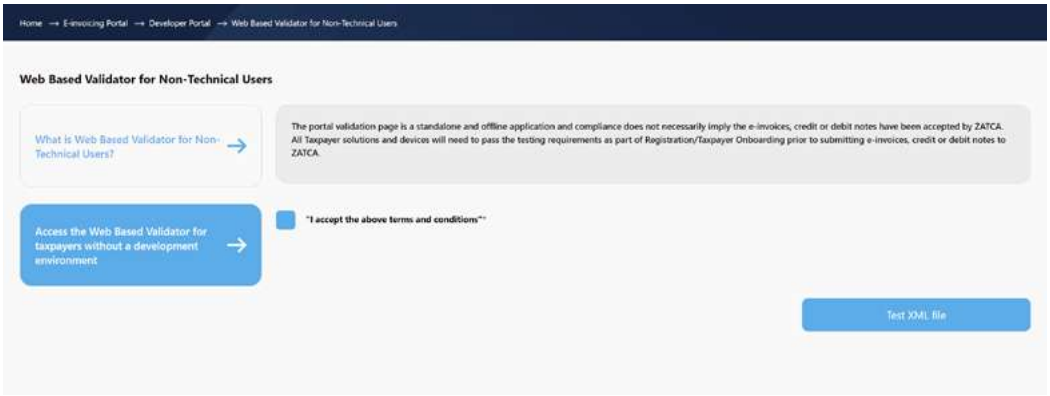


Web based validator Page

- Once users have chosen to "Access the Web Based Validator for taxpayers without a development environment", a disclaimer is shown detailing that:

The portal validation page is a standalone application and compliance does not necessarily imply the e-invoices, credit or debit notes have been accepted by ZATCA. All Taxpayer E-invoicing solution unit will need to pass the testing requirements as part of Registration/Taxpayer Onboarding prior to submitting e-invoices, credit or debit notes to ZATCA.

- The User has to acknowledge the disclaimer in order to proceed to test their XML files.



Web based validator Disclaimer

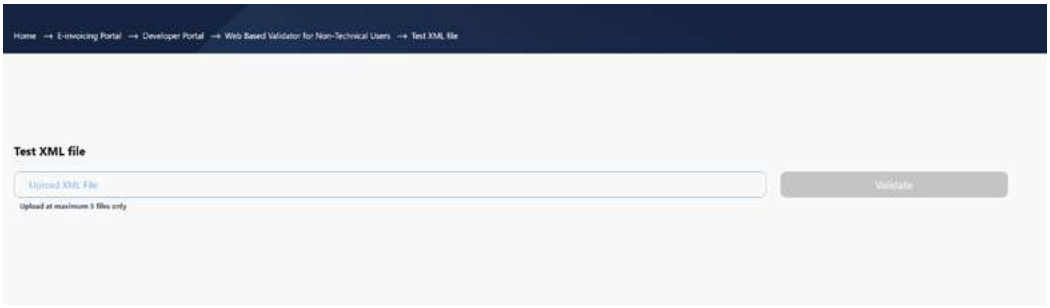
2.3.7 Using the Web Based Validator for Non-Technical Users

An XML file can be validated according to its structure (schema), fields, or ZATCA requirements (i.e. The VAT registration number must be 15 numeric digits). The way this works is that the user submits an XML and the portal will read it, analyze it, and return the status of the validation.

Note that the Web Based Validator can be used to validate up to 5 XMLs and if more than 1 XML is provided, the validator also checks for the sequence in terms of Previous Invoice (Document) Hash. Note that for a single XML the Previous Document Hash check is always considered as valid or True.

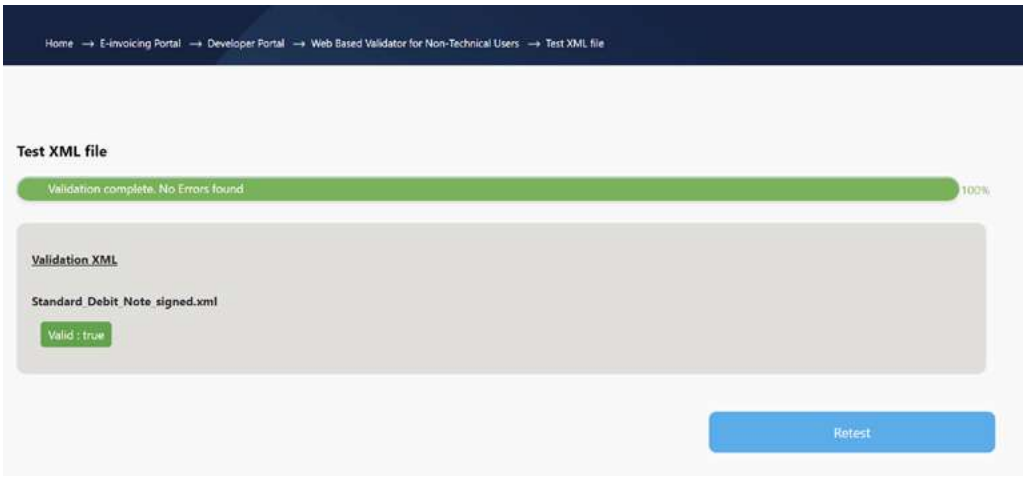
The process for validating XMLs from the Web Based Validator for Non-Technical Users page is as follows:

Click on "Upload XML file" and choose a file, then click "Validate."



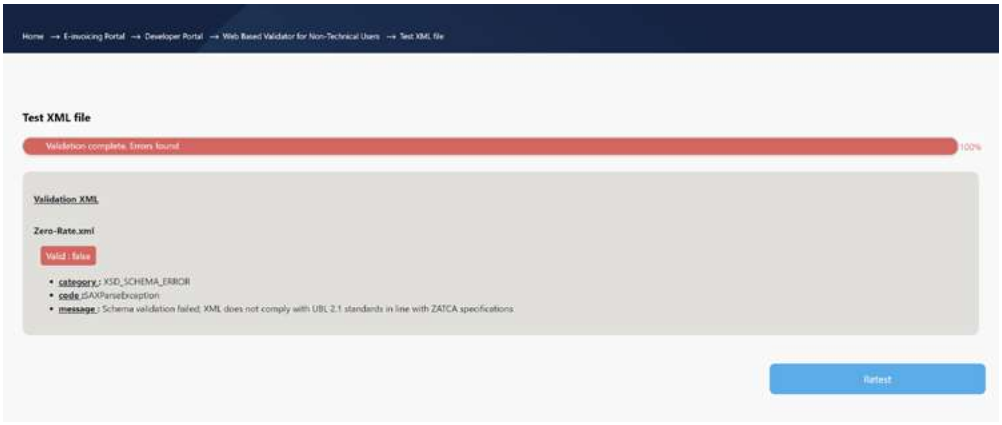
Uploading an XML file on the web based validator

If the XML is compliant, you will receive a "Valid": true message.



Web based validator - XML validation complete and no errors found

If not compliant, the following message is shown.



Web based validator - XML complete and errors found

The non-technical user is expected to share the validation outcomes with the Solution Developer to take necessary action.

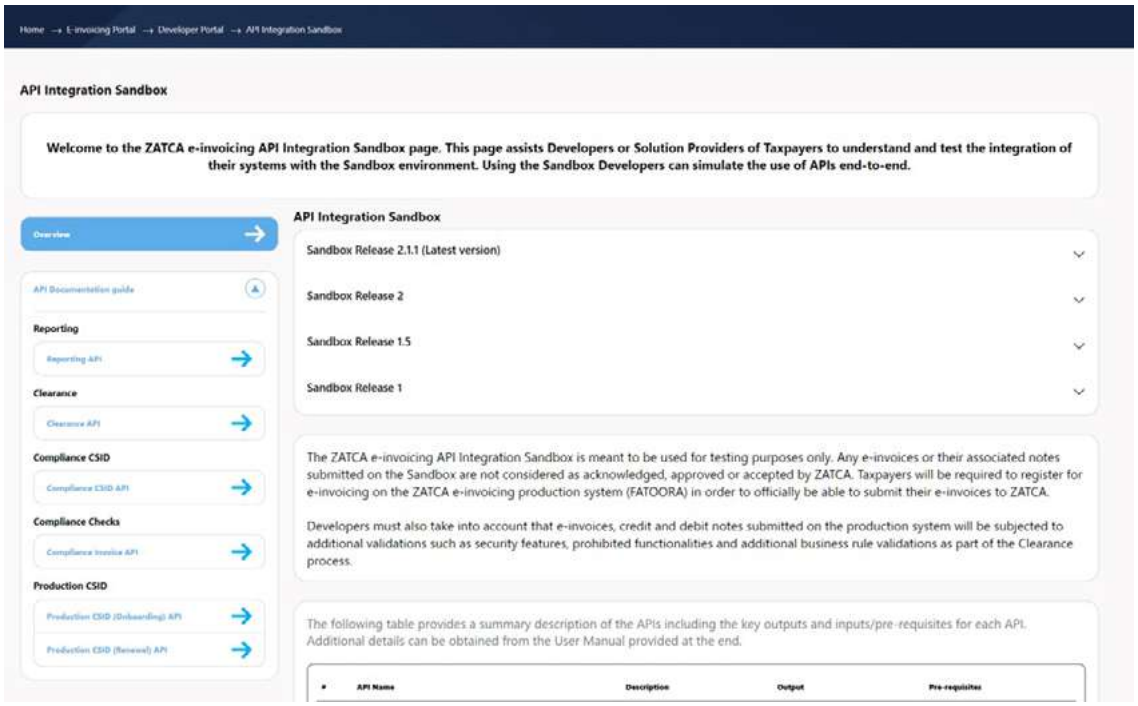
2.3.8 Accessing the Integration Sandbox Page

The Integration Sandbox as covered in this user manual comprises of two components - the Sandbox specific front-end web pages (which is part of the Developer Portal and access to which requires a Developer Portal registered user account) and an API based Sandbox backend to integrate with.

A registered and logged in user can access the Integration Sandbox page from the main dashboard while a non-registered and non-logged in user is taken to the login screen. Once on the Integration Sandbox page the user is given a high level summary of the current version release of the Sandbox as well as links to any previous releases.

The ZATCA e-invoicing integration Sandbox is meant to be used for testing purposes only. Any inputs submitted on the Sandbox are not considered as acknowledged, approved or accepted by ZATCA. Taxpayers will be required to login using SSO credentials for the Taxation portal (ERAD) prior to officially be able to submit official documents. Test CSIDs provided by the Sandbox cannot be used in the Core E-invoicing Solution.

Developers must also take into account that documents or requests submitted on the Core E-invoicing Solution will be subjected to additional validations such as security features, prohibited functionalities, additional business rule validations and/or referential checks based such as validating Seller/Buyer information entered in the documents, validations based on previously submitted documents.



API Integration sandbox landing page

On the left navigation bar of the page the user is able to access the links to the API documentation which are maintained as Swagger files (each API call is described in section 2.3.10 below along with the possible outcomes).

2.3.9 Accessing the API and associated Documentation (Swagger Files)

Access to the Swagger files is provided from the Integration Sandbox page. API documentation is provided covering all the API calls that can be tested on the Sandbox such as:

- Test request for Compliance CSID as part of a new onboarding (requires a signed test CSR to be submitted - details provided in the Swagger files)
- Test request for Production CSID as part of a new onboarding (requires a test Compliance CSID to be submitted)

Note: The Core E-invoicing Solution will require specific compliance checks to be completed in between the Compliance CSID and Production CSID requests and the latter will return an invalid response until these compliance checks are completed. This invalid response can be tested in the Sandbox by providing a specific input which is covered in the Swagger files below.

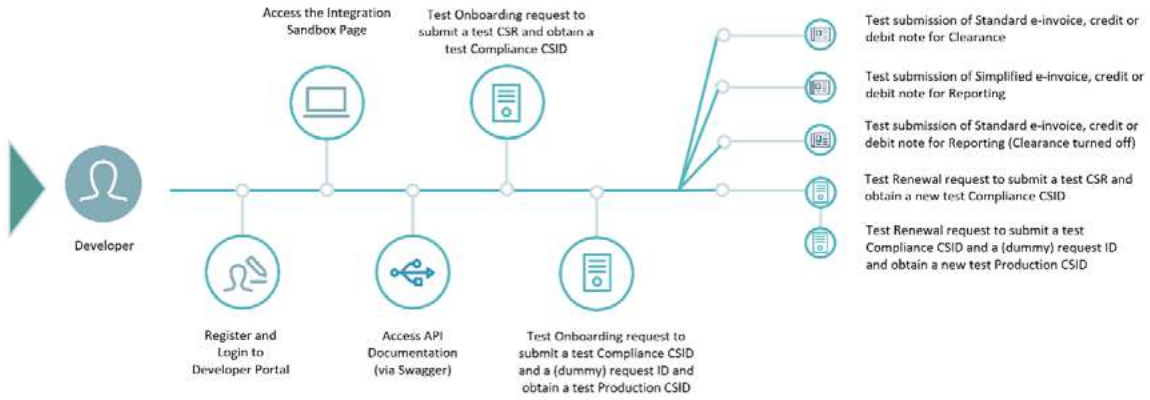
- Test request for a new Production CSID as part of renewal (requires a test Compliance CSID to be submitted)
- Test submission of documents for Clearance (requires a test Production CSID)
- Test submission of documents for Reporting (requires a test Production CSID)

Although the Sandbox uses test CSIDs, it is important to note that the VAT Registration number used to obtain the test CSID must match with the VAT Registration number in the Renewal CSR and/or e-invoices, credit notes, debit notes and QR codes submitted in all subsequent calls made using that specific test CSID. In other words for every VAT Registration Number that is used in the Sandbox integration, a separate CSID will have to be requested. Of course the VAT Registration Numbers can be dummy inputs.

- Please refer to the API Documentation through the following LINK.

Note: Please make sure to log-in in order to view the API documentation

2.3.10 Step by step guide to make a successful call to APIs



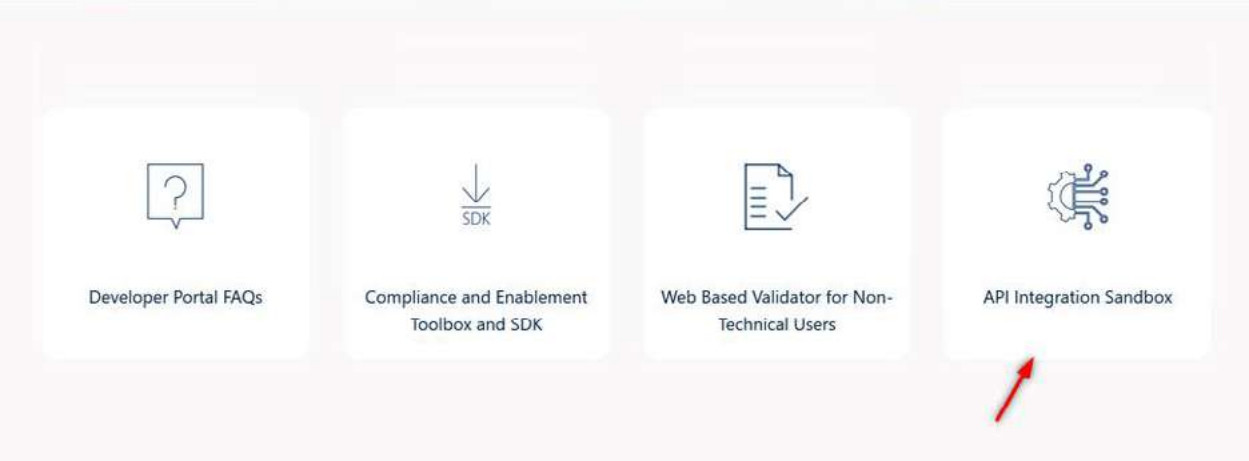
1. For Reporting and Clearance (testing the submission of E-invoices, credit and debit notes)
    - The users' E-Invoice Generation Solution (EGS) needs to generate compliant XML documents. For more details on generating compliant XML documents please refer to the XML Implementation Standards and the Data Dictionary (E-Invoice specifications (zatca.gov.sa). It is also recommend to test the compliance using the Compliance and Enablement Toolbox SDK (Download SDK (zatca.gov.sa) or Portal based validator for non-technical users (Compliance and Enablement Toolbox portal).
    - For Simplified documents (and optionally for Standard documents), the EGS also needs to generate compliant QR codes. For more details on generating compliant QR codes please refer to the Security Features and Implementation Standards (E-Invoice Security Features and Implementation Standards).

- Note that EGS must obtain a test Cryptographic Stamp Identifier (CSID) first, by using the test integration calls for Onboarding or Renewal.

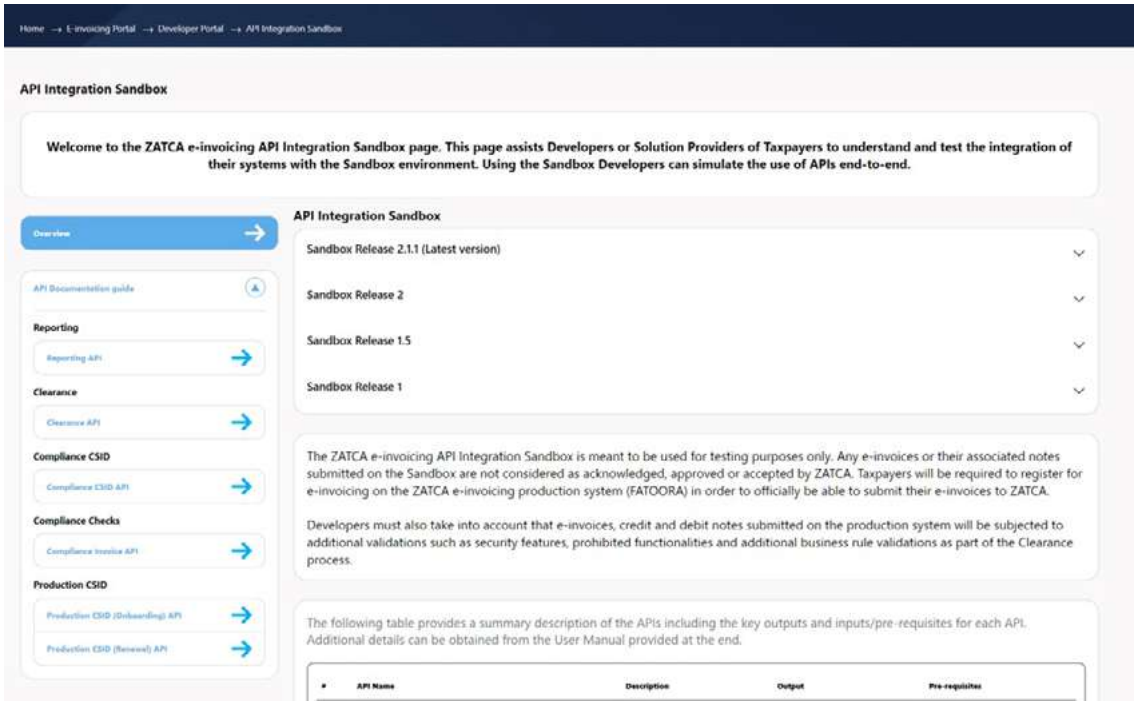
2. For Cryptographic Stamp Identifier (testing the Onboarding and Renewal processes).
    - The users' EGS needs to generate a compliant CSR to obtain a test CSID. For more details on generating a compliant CSR and CSID specifications please refer to (E-Invoice Security Features and Implementation Standards).
    - Note that EGS must obtain a test Cryptographic Stamp Identifier (CSID) first, by using the test integration calls for Onboarding, in order to test the integration call for Renewal which requires a test CSID to be included in the request.

2.3.10.1 Compliance CSID

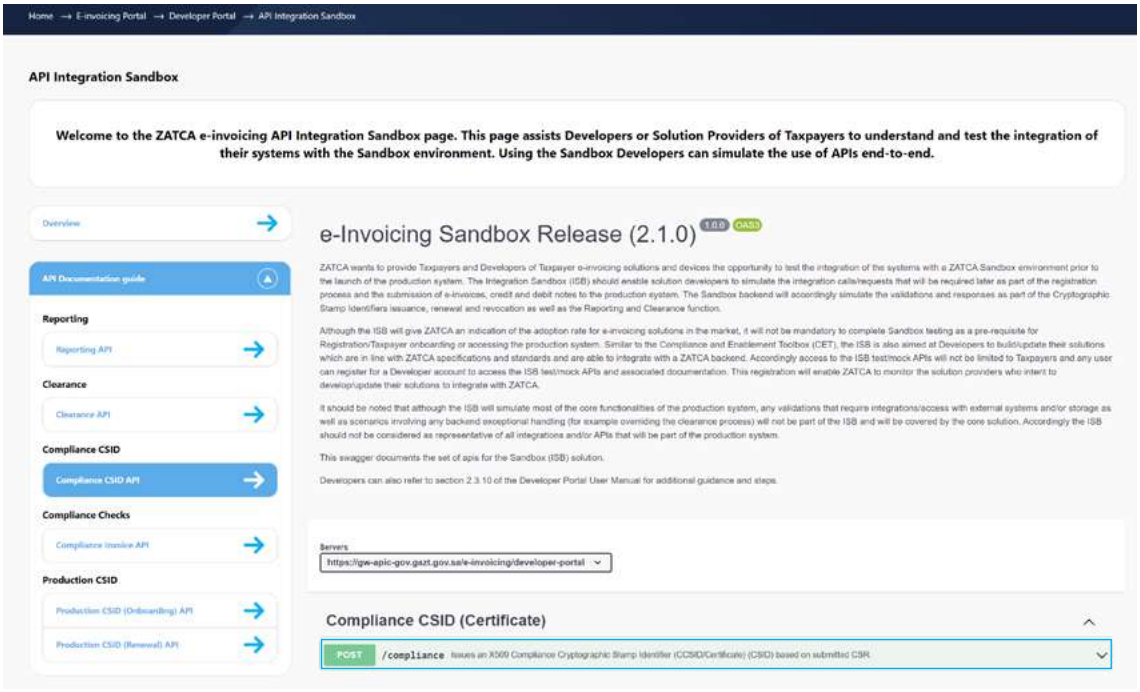
- Step 1: Navigate to Developer Portal link
- Step 2: Login with correct credentials
- Step 3: Navigate to API Integration Sandbox



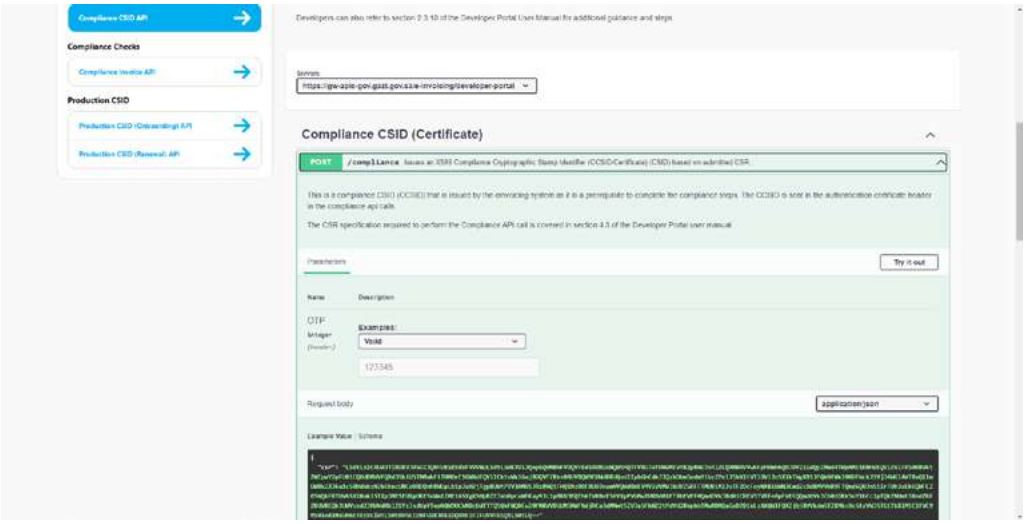
- Step 4: Click on API documentation guides



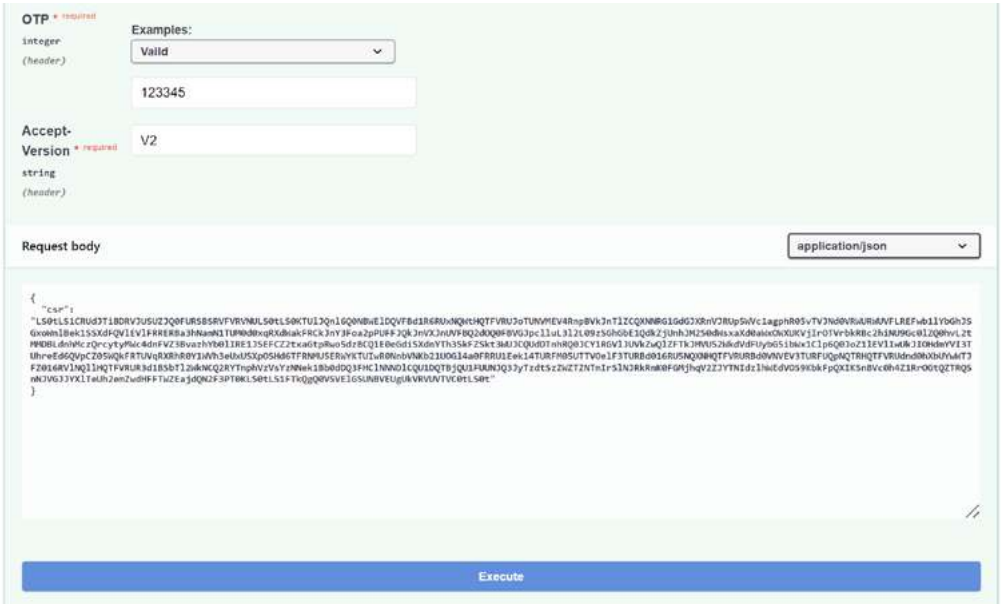
- Step 5: Click on Compliance CSID API drop-down



- Step 6: Click on try it out

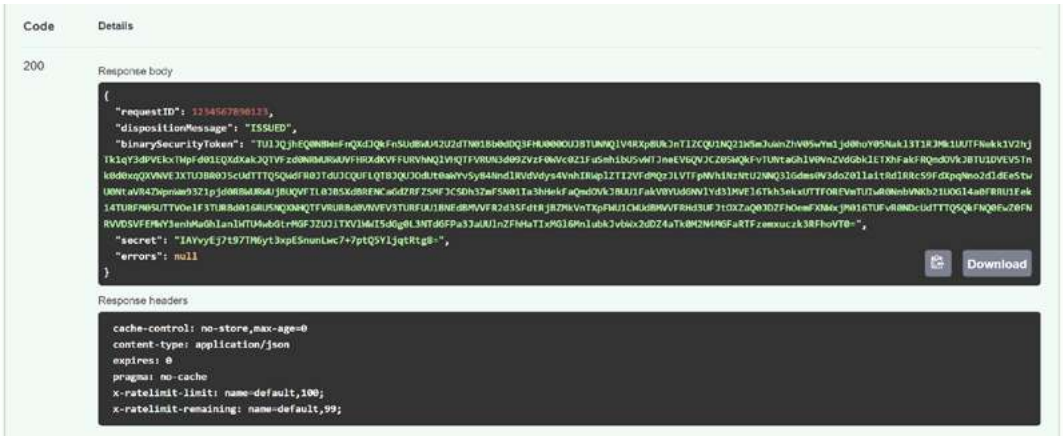


- Step 7: Insert valid OTP and fill the request body (CSR)
- Step 8: Click on Execute



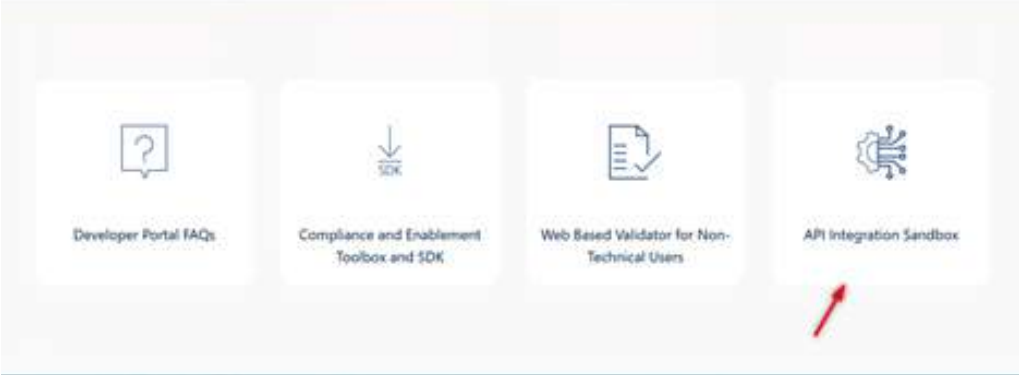
Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

- Result (200)

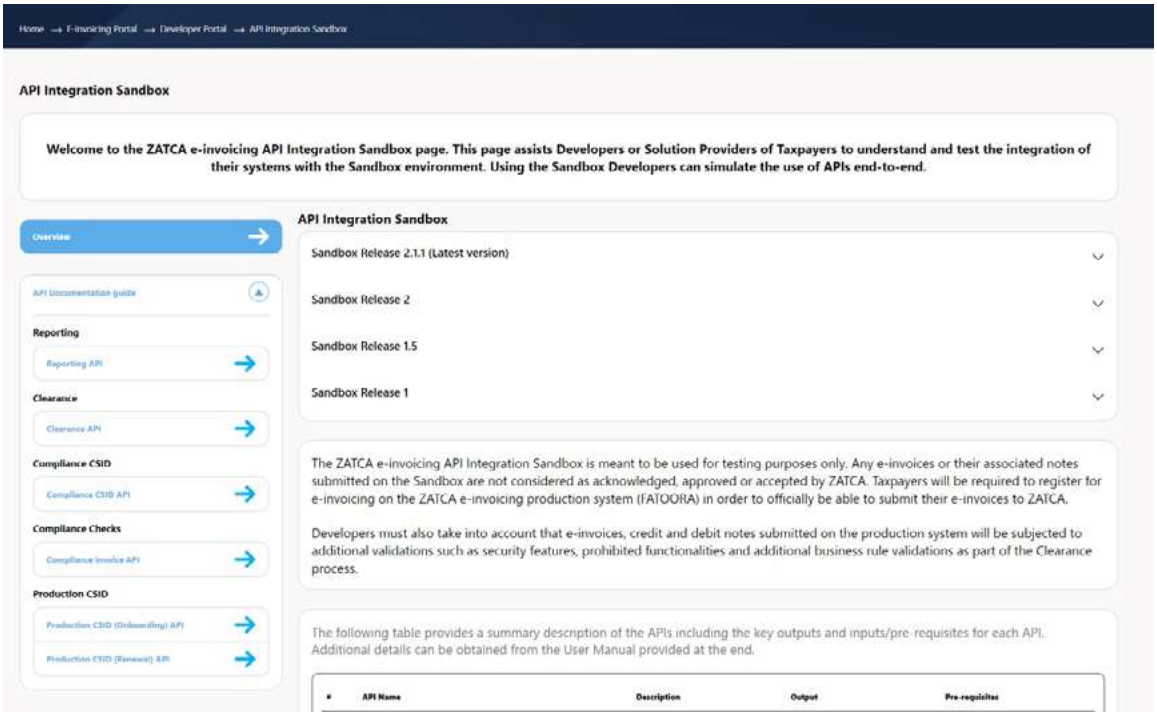


2.3.10.2 Compliance Invoice API

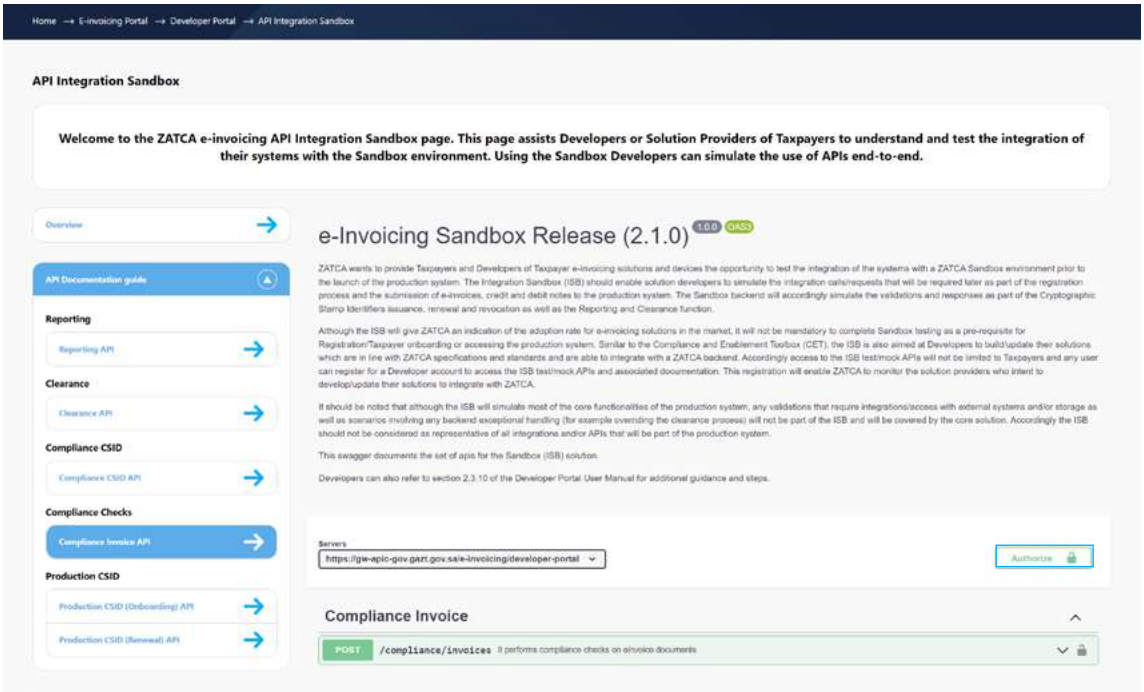
- Step 1: Navigate to Developer Portal link
- Step 2: Login with correct credentials
- Step 3: Navigate to API Integration Sandbox



- Step 4: Click on API documentation guide



- Step 5: Click on Authorize Compliance Invoice API



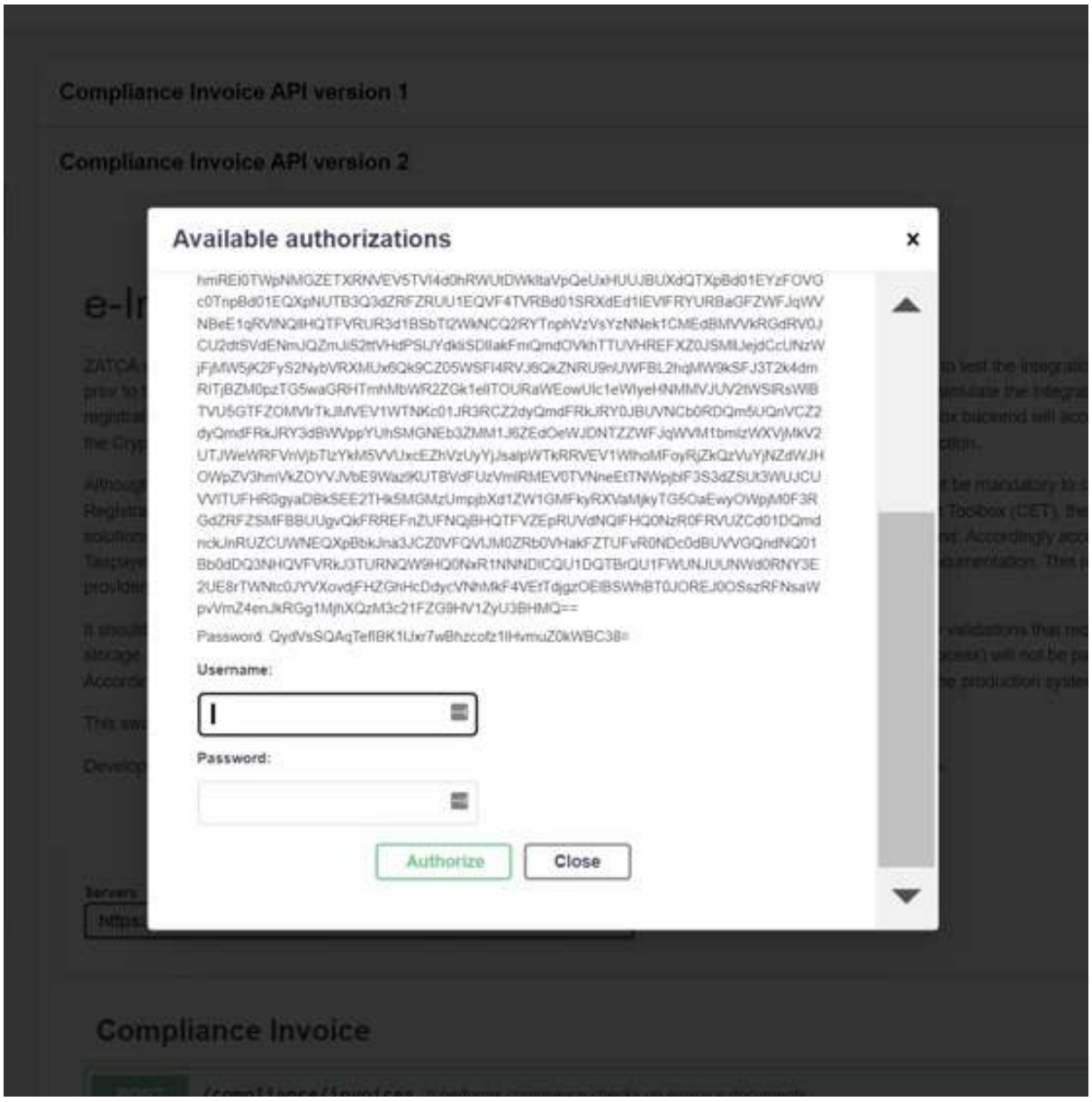
Note: This step is to be repeated on the number of invoices to be sent as part of the compliance checks.

- Step 6:

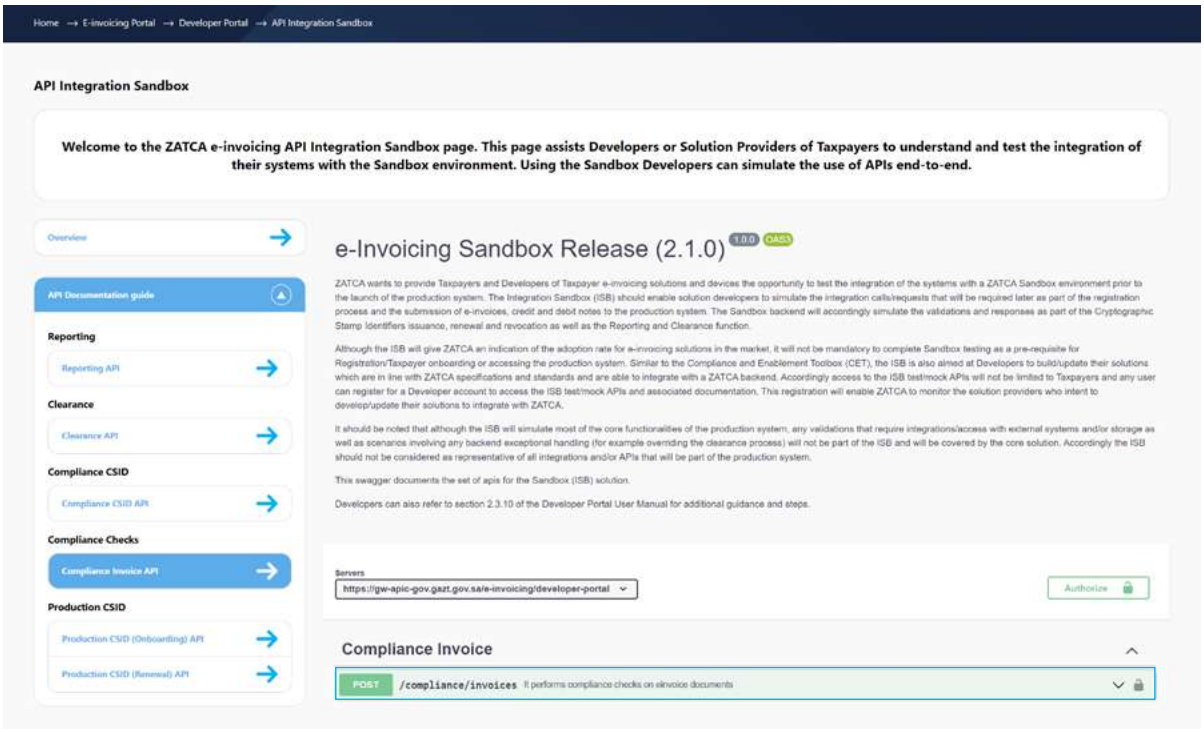
For the Sandbox, use the sample dummy Username and Password provided to you on the Authorization screen.

For Production, run the Compliance CSID API to obtain the "binarySecurityToken" to be used as the Username and "secret" as the Password.

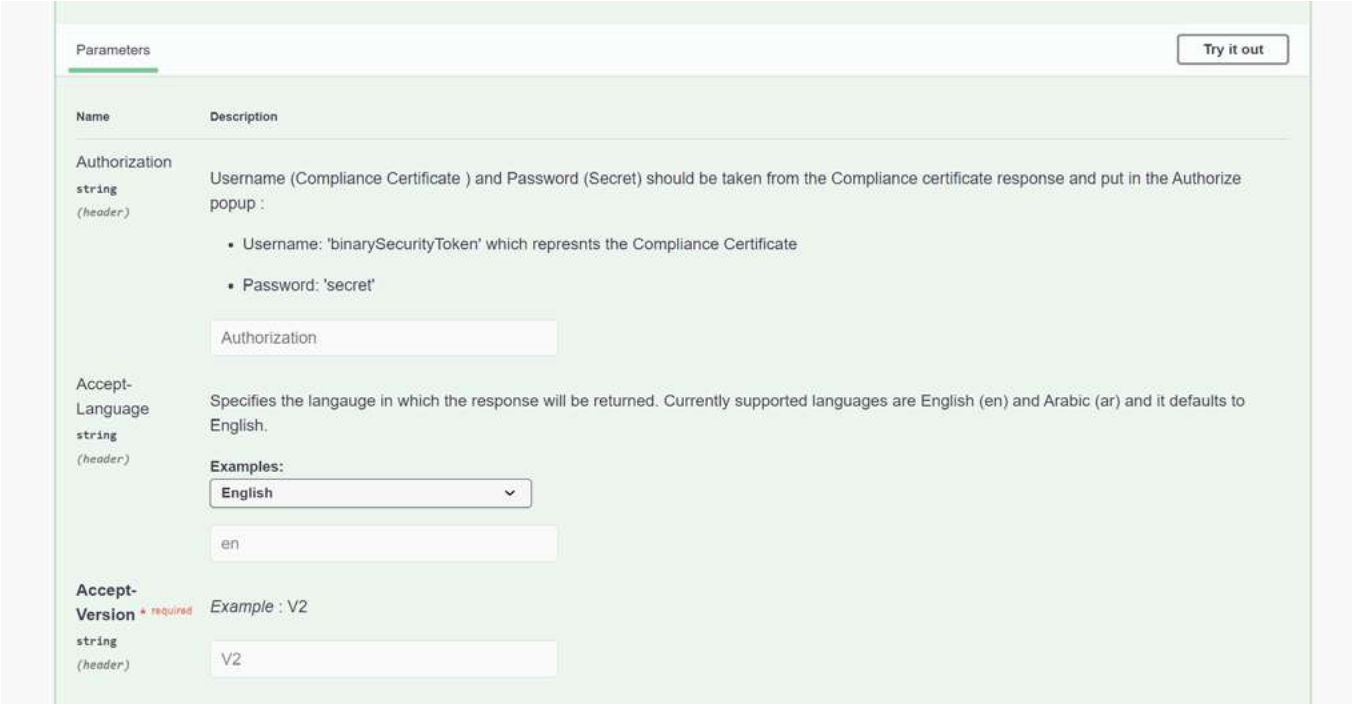
Please refer to Chapter 3 of this document for further details.



- Step 7: Click on Compliance Invoice API drop-down

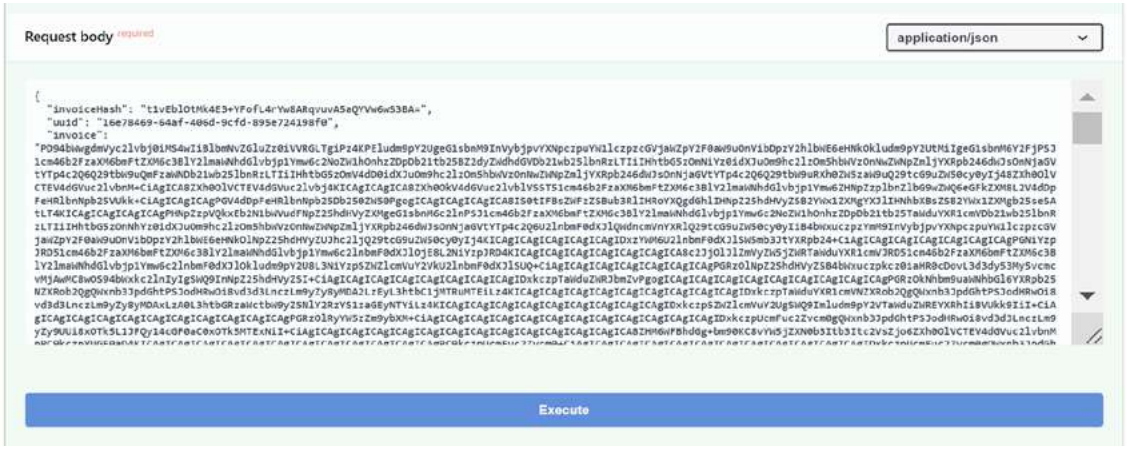


- Step 8: Click on Try it out button

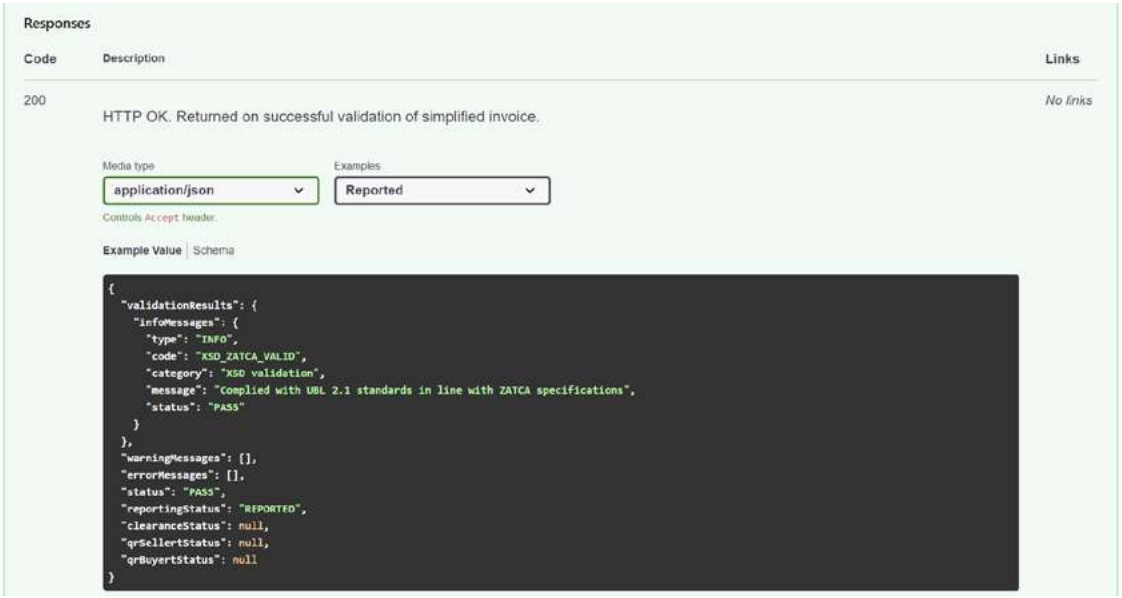


Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

- Step 9: Fill the request body (invoice hash, UUID, encoded XML invoice)
- Step 10: Click on Execute



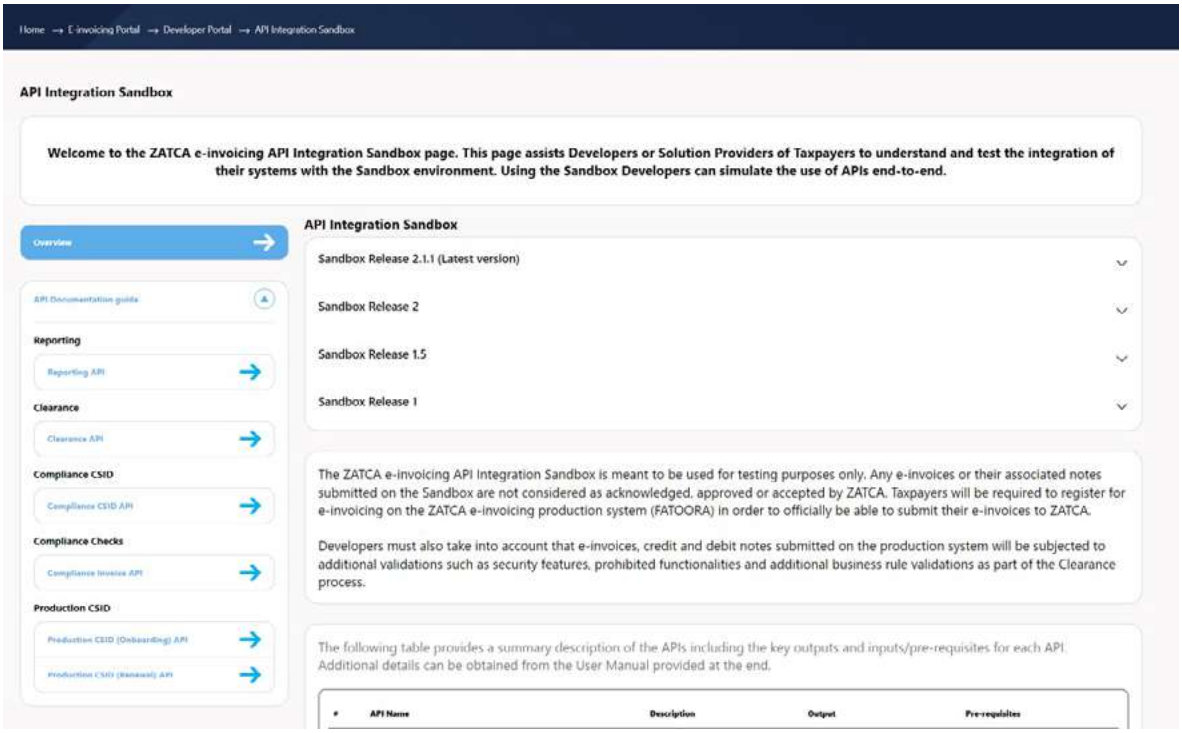
- Result (200)



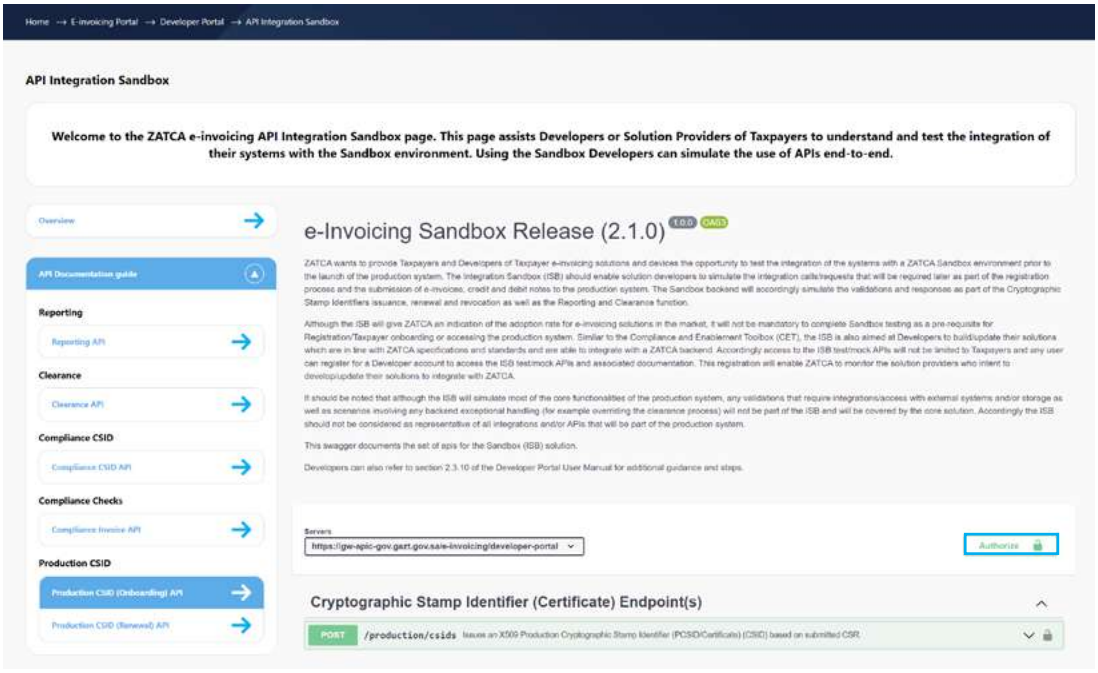
2.3.10.3 Production CSID (Onboarding) API

- Step 1: Navigate to Developer Portal link
- Step 2: Login with correct credentials
- Step 3: Navigate to API Integration Sandbox

- Step 4: Click on API documentation guide



- Step 5: Click on Authorize Production CSID (Onboarding) API

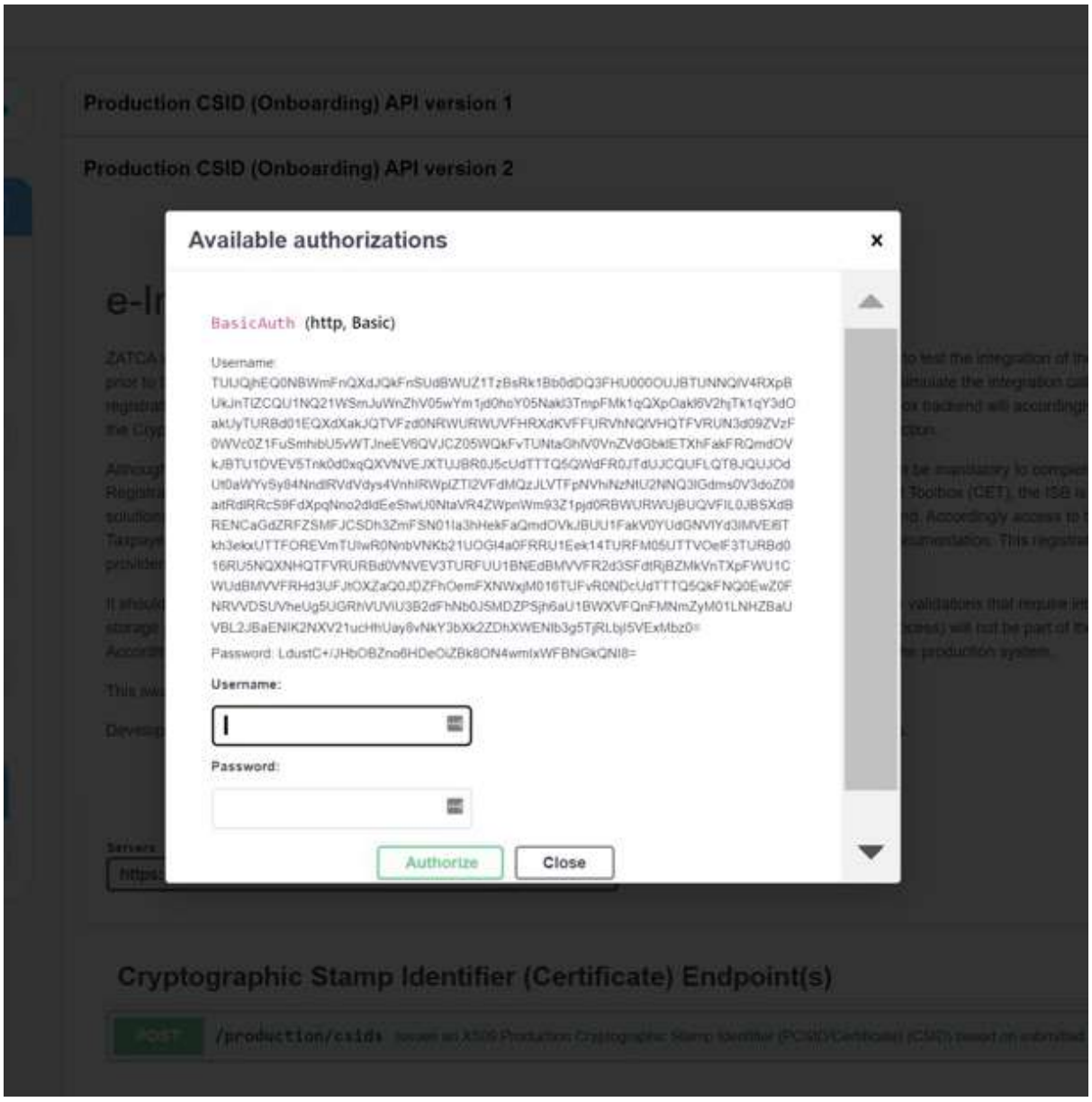


- Step 6:

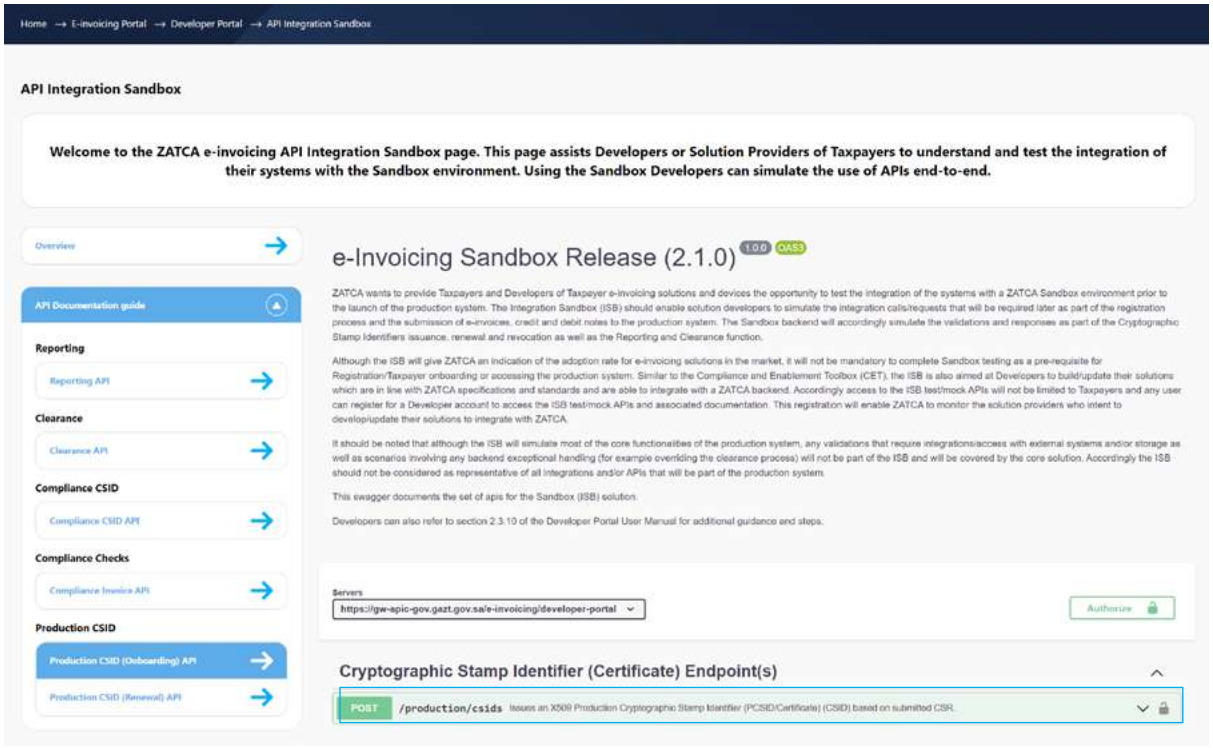
For the Sandbox, use the sample dummy Username and Password provided to you on the Authorization screen.

For Production, run the Compliance CSID API to obtain the "binarySecurityToken" to be used as the Username and "secret" as the Password.

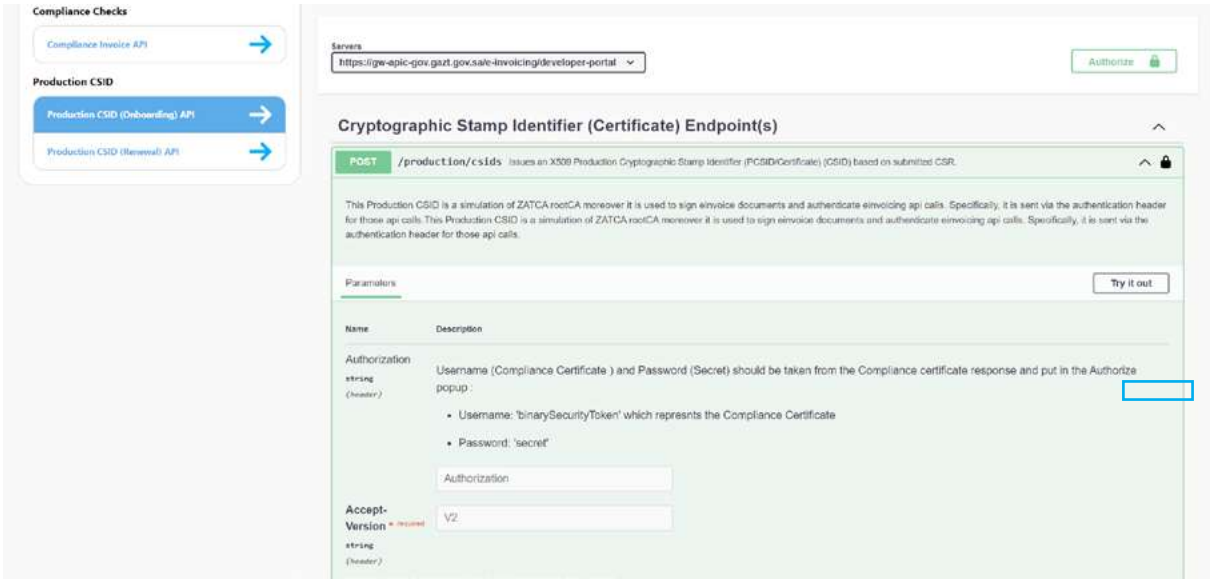
Please refer to Chapter 3 of this document for further details.



- Step 7: Click on Production CSID (Onboarding) API drop-down

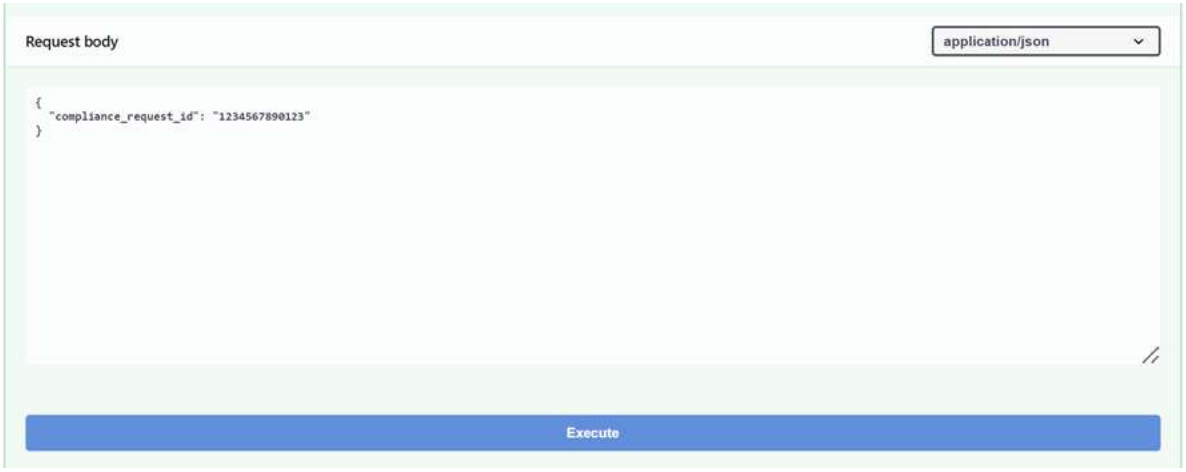


- Step 8: Click on Try it now button

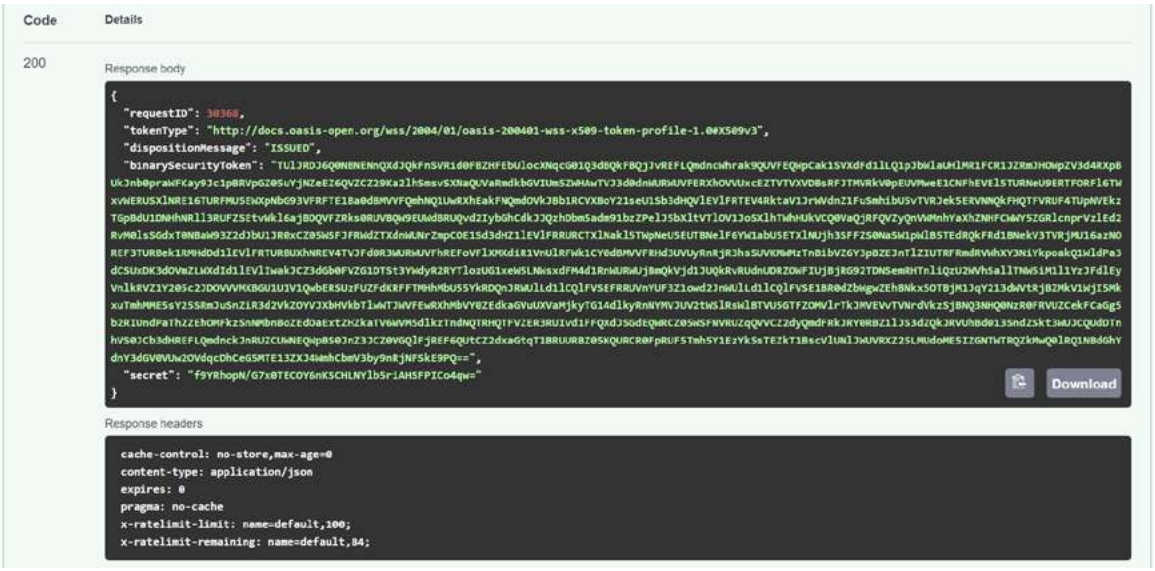


Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

- Step 9: Fill the request body (compliance request ID)
- Step 10: Click on Execute button

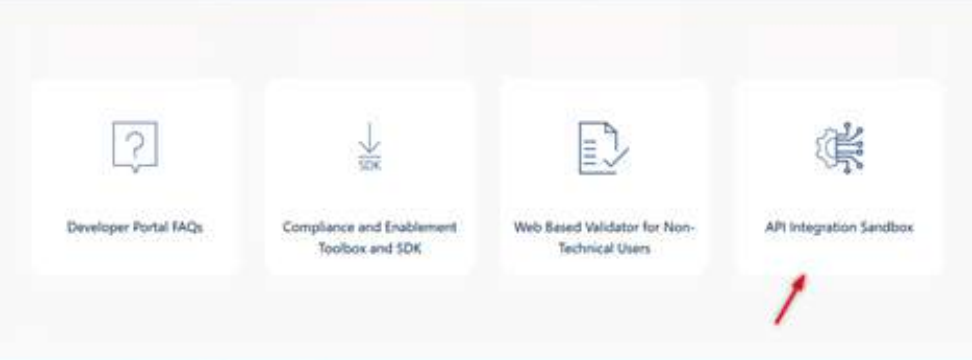


- Result (200)

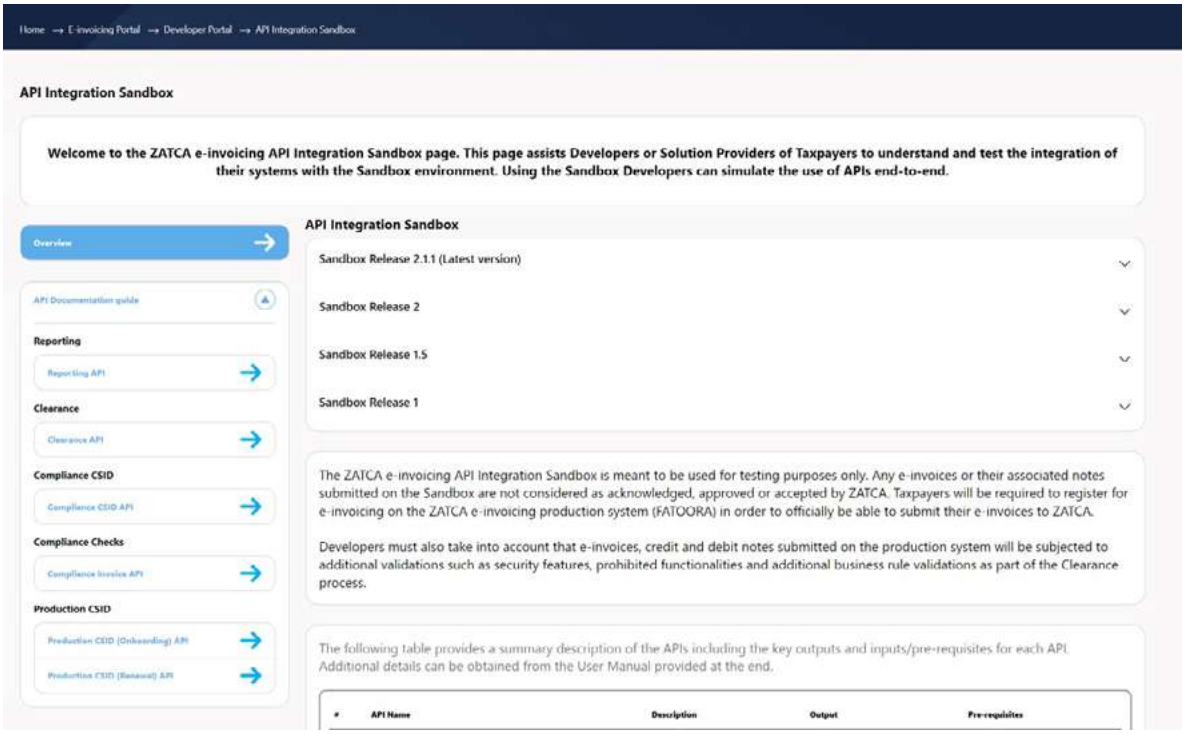

Code
Details
200
Response body
{
"requestID": 30360,
"tokenType": "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3",
"dispositionMessage": "ISSUED",
"binarySecurityToken": "TUlJRDJ6Q0NBNENnQXdJQkFnSVR1d0FBZHFEbUlocXNqcG01Q3dBQkFBQjJvREFLQmdncWhrak9QUVFEQWpCak1SYXdFd1lLQ1pJbWlaUHlMR1FCR1JZRmJHOWpZV3d4RXpB
UkJnb0praWFKay9Jc1pBRVpGZ0SuYjNZeEZ6QVZCZ29Ka2lhSmsvSXNaQUVaRmdkbGVIUm5ZWHAwTVJ3d0dnWURWUVFERXhoVVUxcEZTVTVXVDBsRFJTMVRkV0pEUVMweE1CNFhEVEl5TURNeU9ERTFORFl6TW
xvWERUSXlNRE16TURFMU5EWXpNbG93VFRFTE1Ba0dBMVVFQmhNQ1UwRXhEakFWQmdOVkJBb1RCVXBoY21seU1SbGdhQVlEVlFRTEV4RktaV1JrWVdnZ1FuSmhibU5vTVRJek5ERVNNQkFHQTFVRUF4TUpNVEkz
TGpBdU1DNHhNRll3RUFZSEtvWkl6ajBDQVFZRks0RUVBQW9EUWdBRUQvd2IybGhCdkJQZzhObm5adm91bzZPel5bXltVTlOV1JoSXlhTWhHUkVCQ0VaQjRFQVZyQnVWMnhYaXhZNHFCWWY5ZGRlcnprVzlEd2
RvM0lsSGdxT0NBaW93Z2djbU1JR0xCZ05WSFJFRWdZTXdnWUNrZmpCOE1Sd3dHZ1lEVlFRRURCTXlNak1STWpNeU5EUTBNelF6YW1abU5ETXlNUjh3SFFZS0NaSW1pWlB5TEdRQkFRd1BNekV3TVRjMU16azNO
REF3TURBek1RMHdDd1lEVlFRTURBUXhNREV4TVJFd0R3WURWUVFhREFoVFlXMXdi01VnUlRFWk1CY0dBMVVFRHd3UVUyRnRjR3hsSUVKMWMzTnBibVZGY3pBZEJnTlZIUTRFRmRVWhXY3NiYkpoakQ1WldPa3
dCSUxDK3dOVmZLWXdId1lEVlIwakJCZ3dGb0FVZG1DTSt3YWdyR2RYTlozUG1xeW5LNWsxdFM4d1RnWURWUjBmQkVjdlJUQkRvRUdnUDRZOWFIUjBjR69ZTDNSemRHTnliQzU2WVhsallTNW5iMll1YzJFdlEy
VnlkRVZ1Y205c2JDOVVVMXBGU1U1V1QwbERSUzFUZFdKRFFTMHhMbU55YkRQnJRWUlldllCQlFVSEFRRUVnYUF3Z1owd2JnWUlldllCQlFVSE1BR0dZbWgwZEh8NkxSOTBjM1JqY213dWVtRjB2MkV1WjI5Mk
xuTmhMME5sY25SRmJuSnZiR3d2VkZOYVJXbHVkbTlwNTJWVFEwRXhMbVY0ZEdkaGVuUXVaMjkyTG14dlkyRnNYMVJUV2tWSlRswlBTVU5GTFZOMVlrTkJMVEVVTVNrdVkzSjBNQ3NHQ0NzR0FRVUZCekFCaGg5
bzRIUndPaTh2ZEhOMFkzSnNMbnBoZEdOaExtZHZkaTV6WVM5dlkzTndNQTRMQTFVZER3RUIvd1FFQXdJSGdEQWRCZ05WSFNVRUZqQVVCZ2dyQmdFRkJRY0RBZ1lJS3dzQkJRVUhBd013SndzSkt3WUJCQUdDTn
hvS0JCb3dHREFLQmdncknRU2CUWNEQWpBSOJnZ3JCZ0VGQlFjREF6QUtCZ2dxaGtqT1BBUURBZ05KQURCR0FpRUF5Tmh5Y1EzYk5sTEZkT1BscVlUNlJwUVRXZ2SLMUdoME5IZGNTWTRQZkMwQ0lRQ1NBdGhY
dnY3dGV0VUw2OVdqcDhCeGSMTE13ZXJ4WmhCbmV3by9nkjNFSkE9PQ==",
"secret": "f9YRhopN/G7x0TECOY6nK5CHLNYlb5riAHSFPICo4qw="
}
Download
Response headers
cache-control: no-store,max-age=0
content-type: application/json
expires: 0
pragma: no-cache
x-ratelimit-limit: name=default,100;
x-ratelimit-remaining: name=default,84;

2.3.10.4 Production CSID (Renewal) API

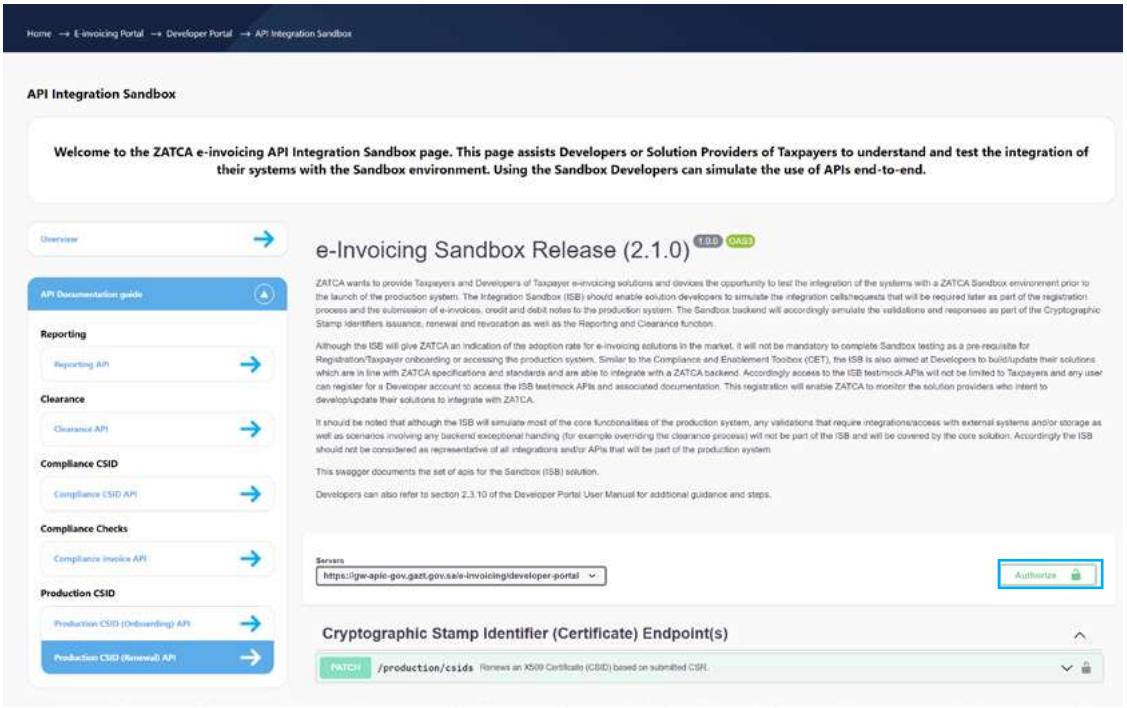
- Step 1: Navigate to Developer Portal link
- Step 2: Login with correct credentials
- Step 3: Navigate to API Integration Sandbox



- Step 4: Click on API documentation guide



- Step 5: Click on Authorize Production CSID (Renewal) API

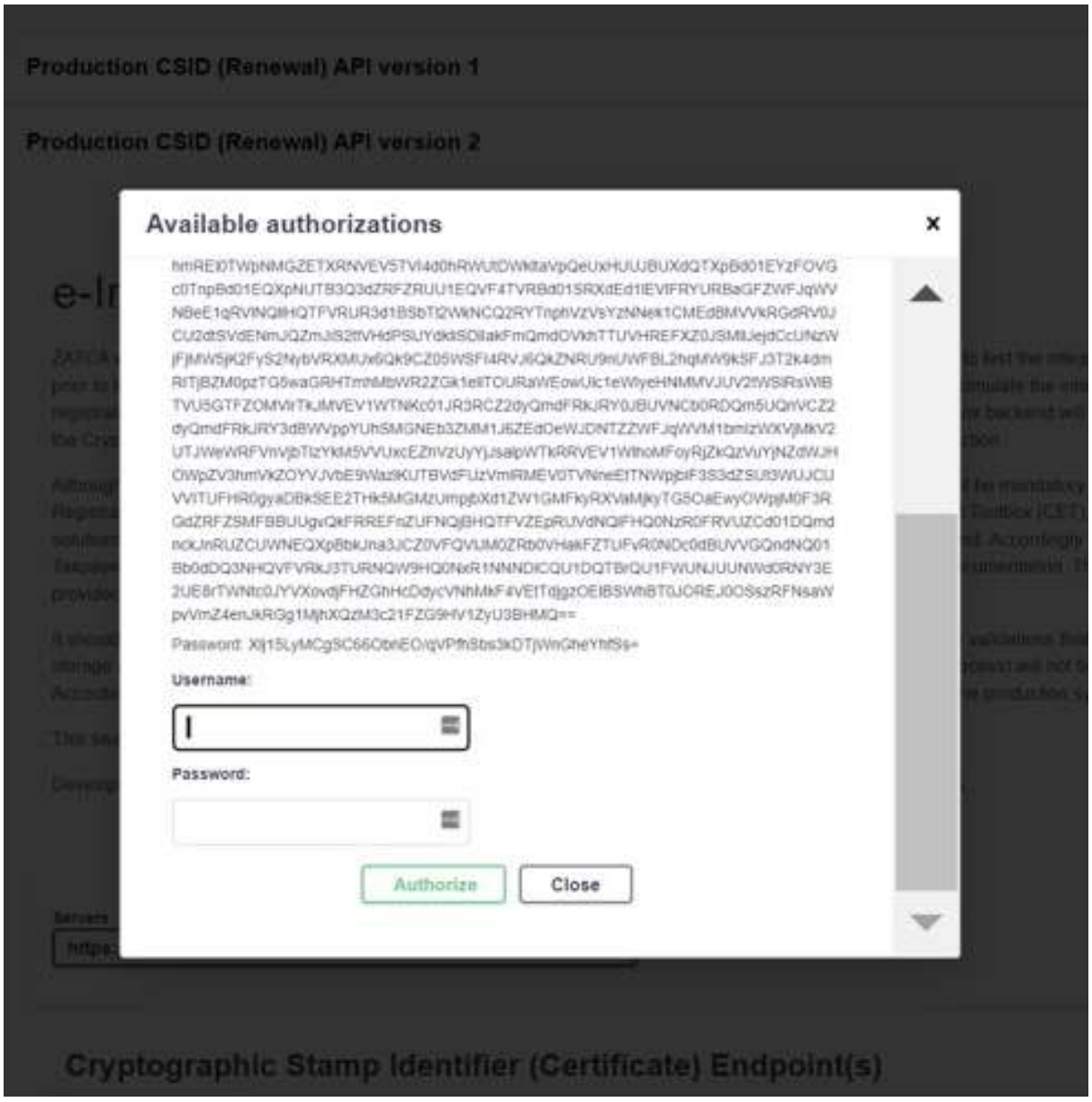


- Step 6:

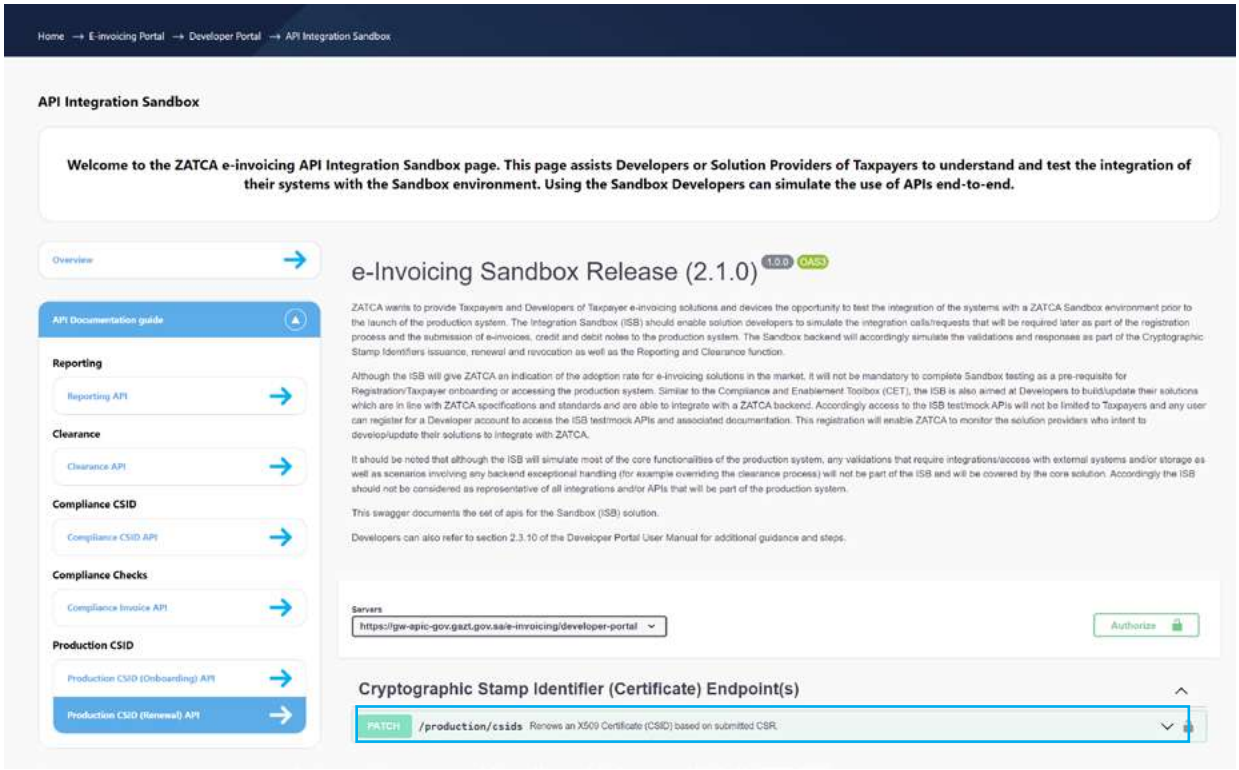
For the Sandbox, use the sample dummy Username and Password provided to you on the Authorization screen.

For Production, run the Compliance CSID API to obtain the "binarySecurityToken" to be used as the Username and "secret" as the Password.

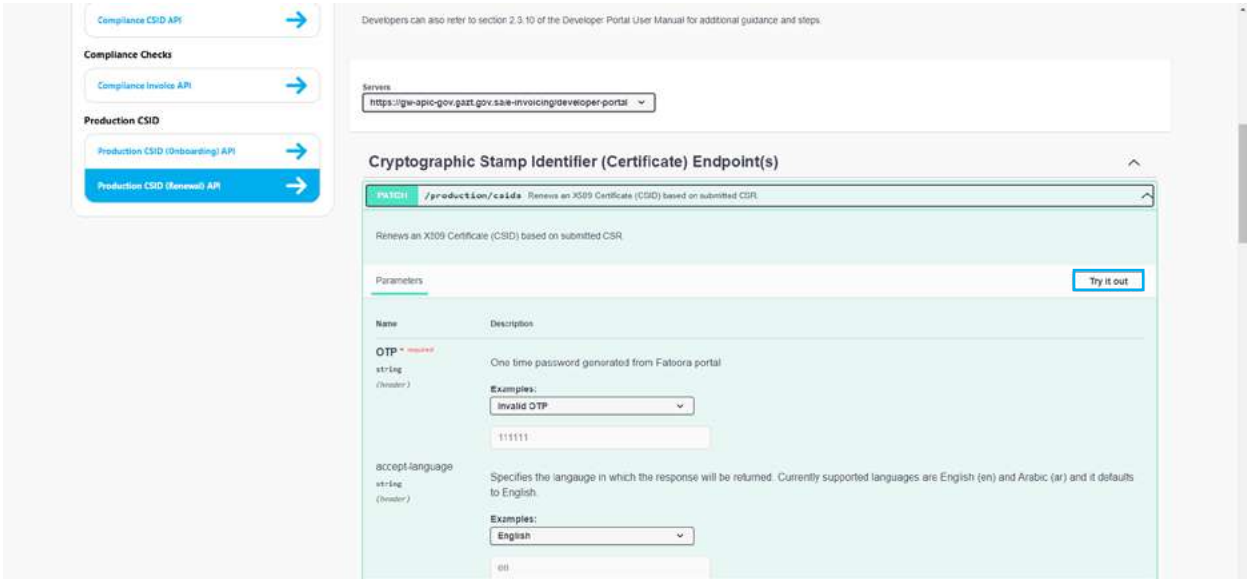
Please refer to Chapter 3 of this document for further details.



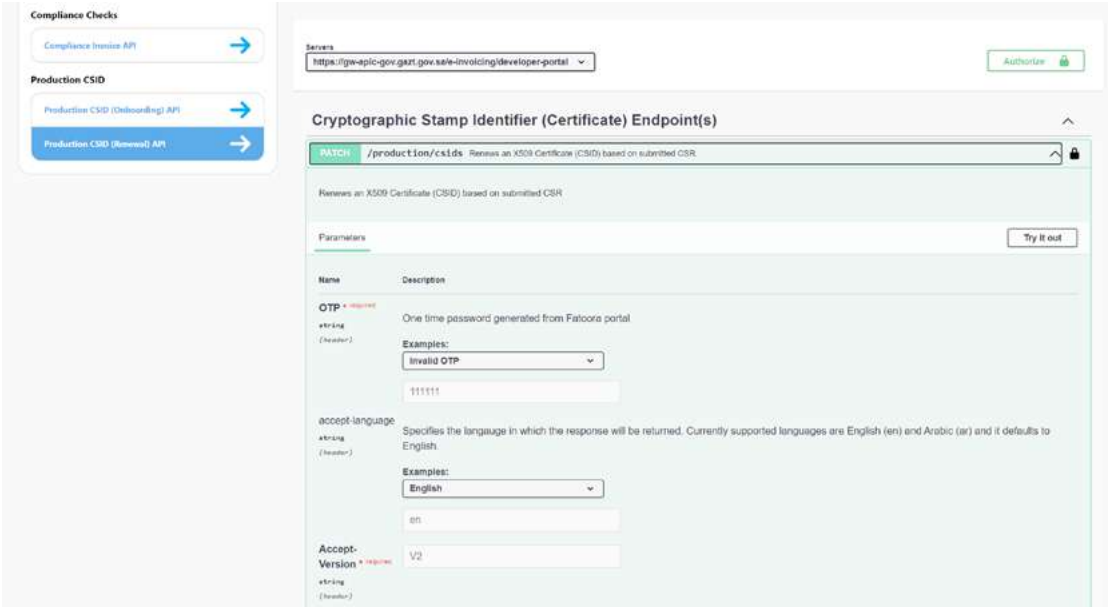
- Step 7: Click on Production CSID (renewal) API drop-down



- Step 8: Click on Try it now

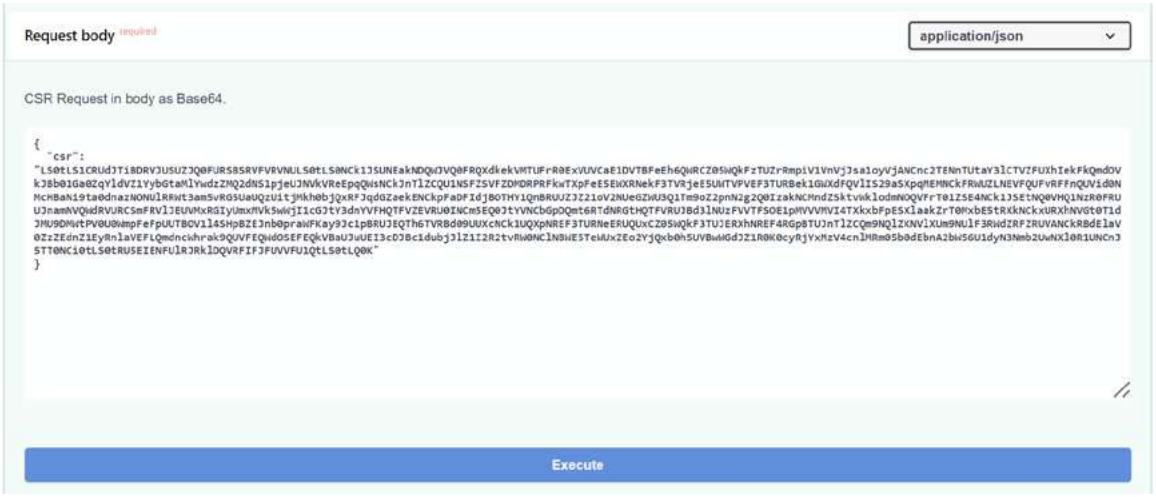


- Step 9: Insert valid OTP

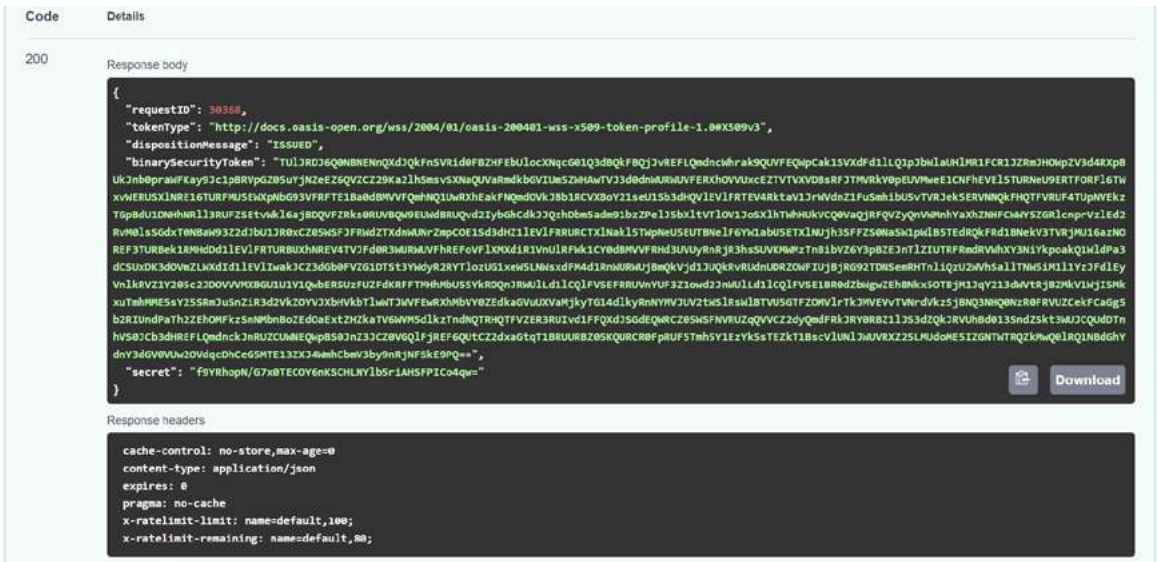


Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

- Step 10: Fill the request body (CSR)
- Step 11: Click on Execute button



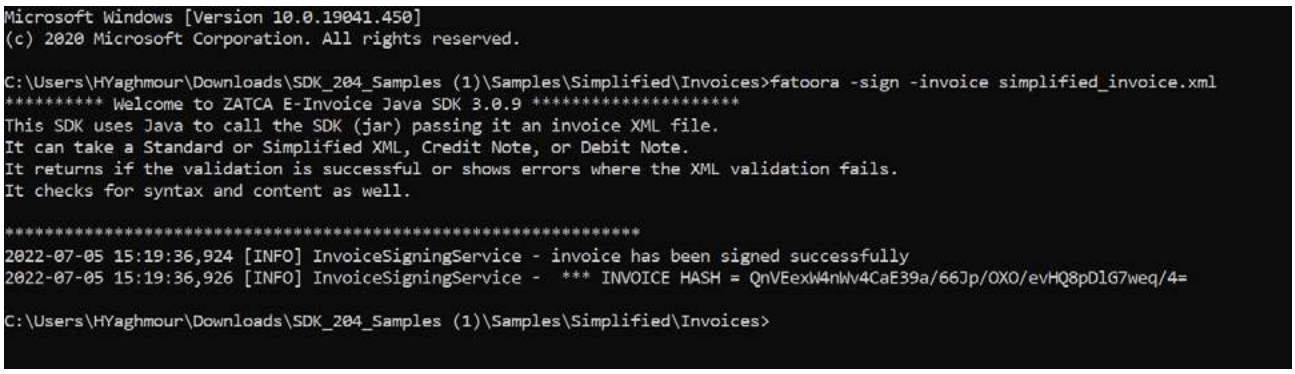
- Result (200)

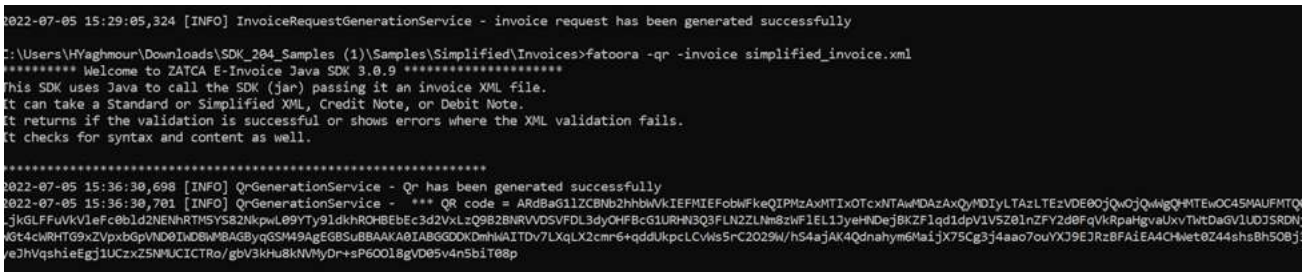


2.3.10.5 REPORTING

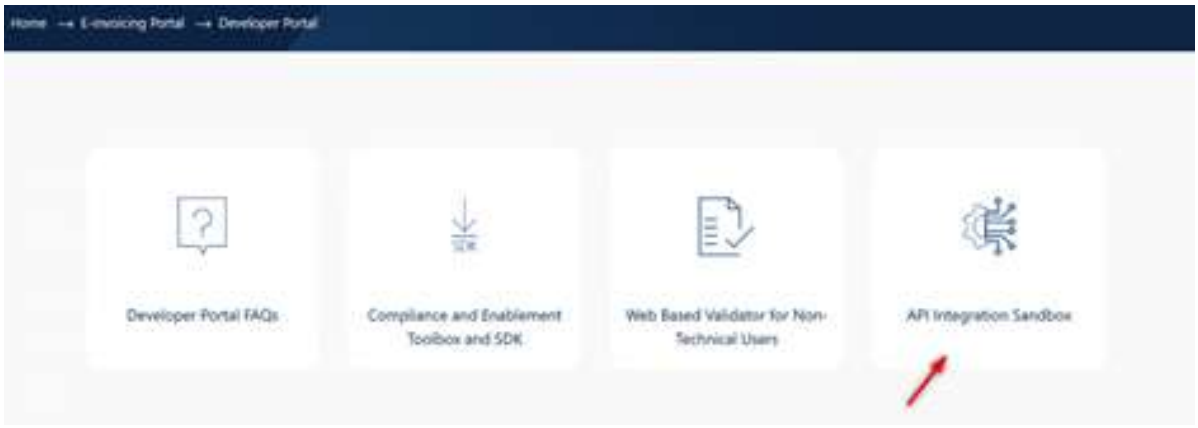
- Step 1: Open CMD and Generate simplified invoice
- Step 2: On SDK, sign the XML invoice and get the hash value using the following:
  - fatoora -sign -qr -invoice invoiceName.xml -signedinvoice signedinvoiceName.xml (hash is returned and signed invoice is generated)
  - fatoora -generateHash -invoice invoiceName.xml (optional)

    Afterwards, validate the XML invoice.

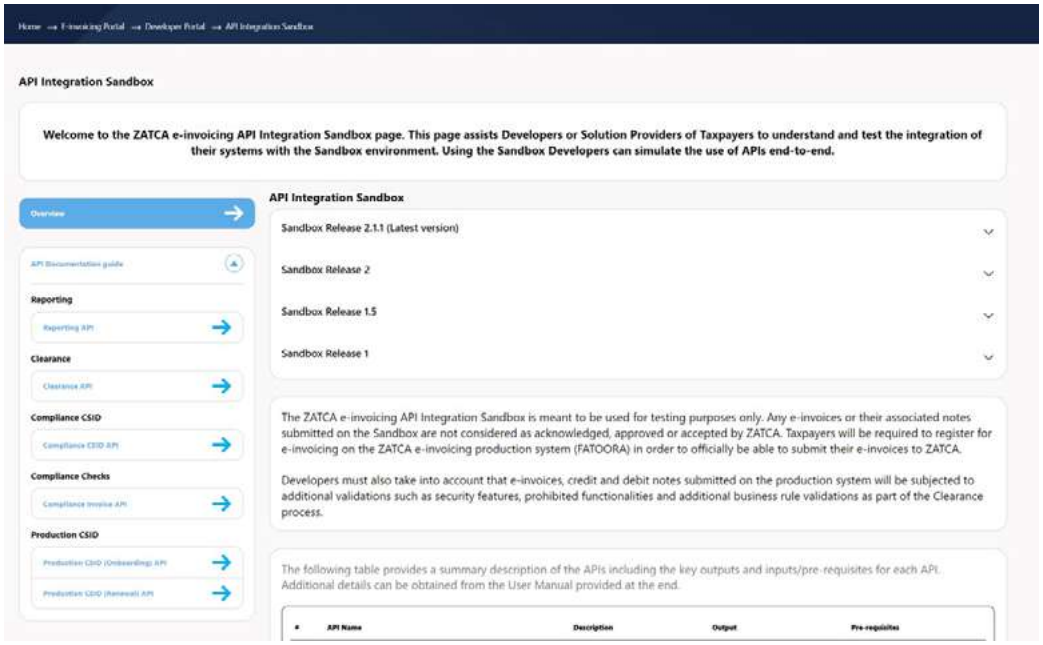




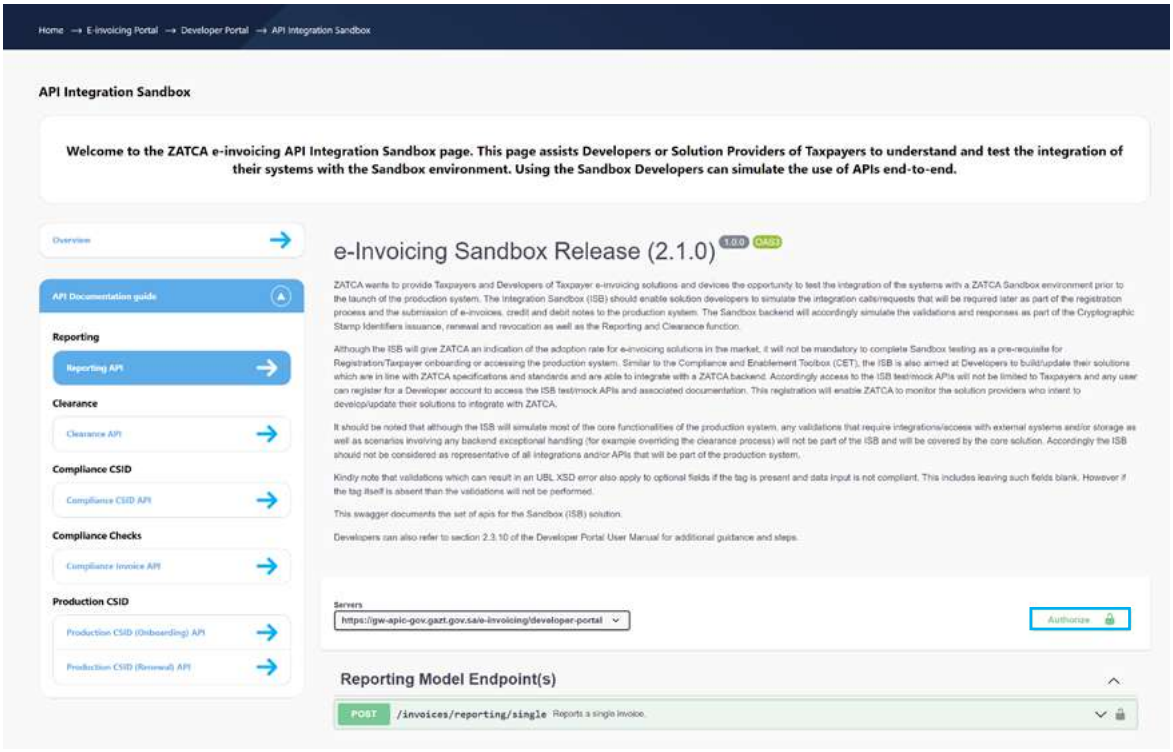
- Step 3: Encode the Signed XML invoice using Base 64
- Step 4: Open Developer Portal and choose integration sandbox



- Step 5: Click on API documentation guides



- Step 6: Click on Authorize Reporting API

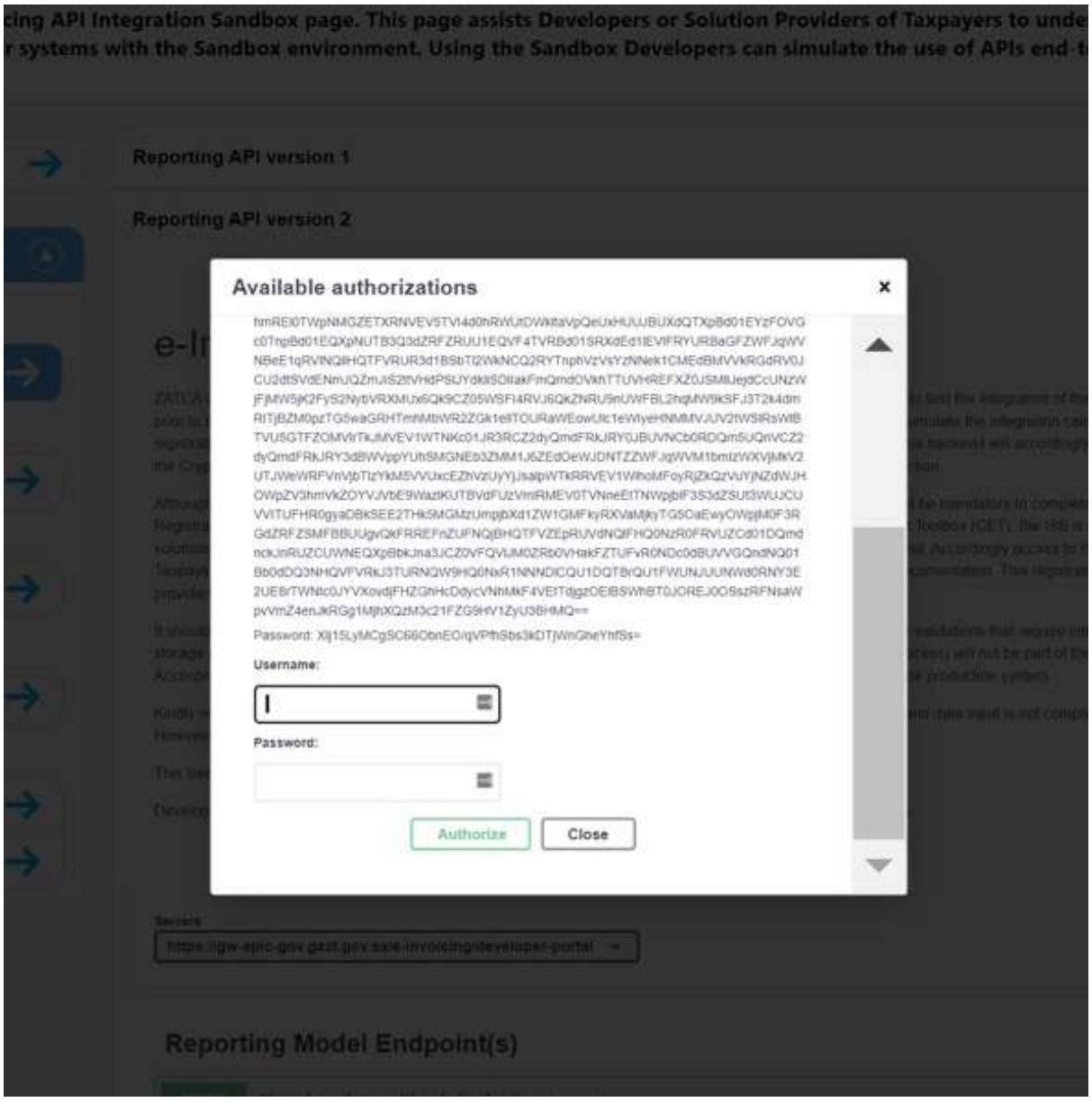


- Step 7:

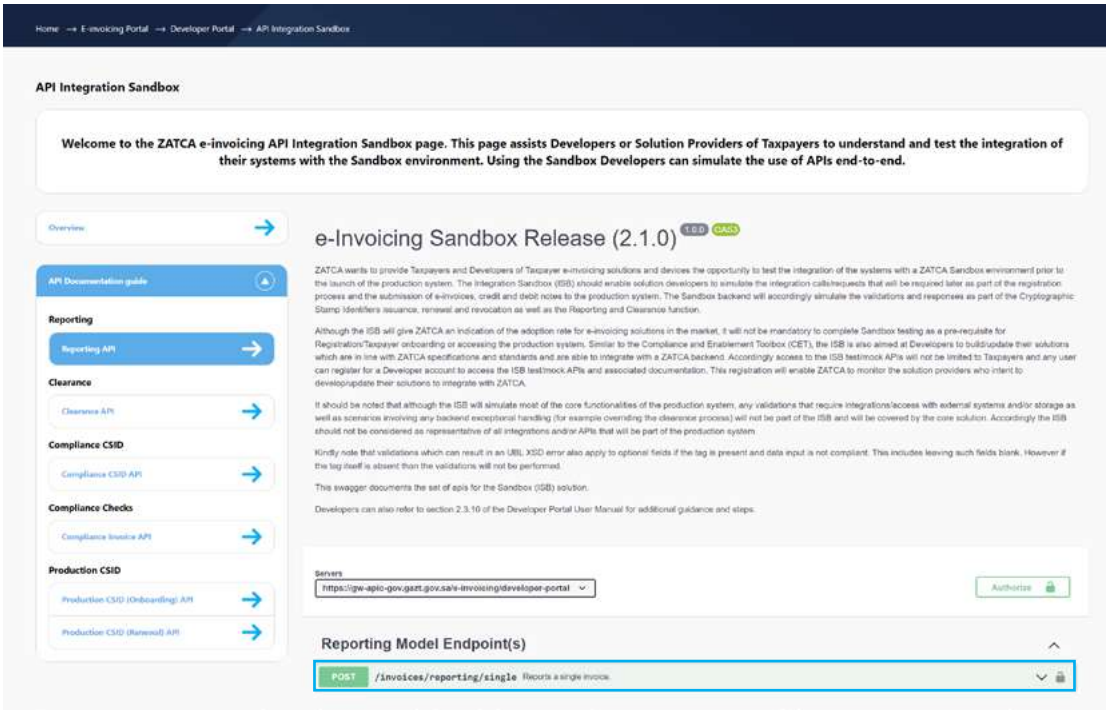
  For the Sandbox, use the sample dummy Username and Password provided to you on the Authorization screen.

  For Production, run the Compliance CSID API to obtain the "binarySecurityToken" to be used as the Username and "secret" as the Password.

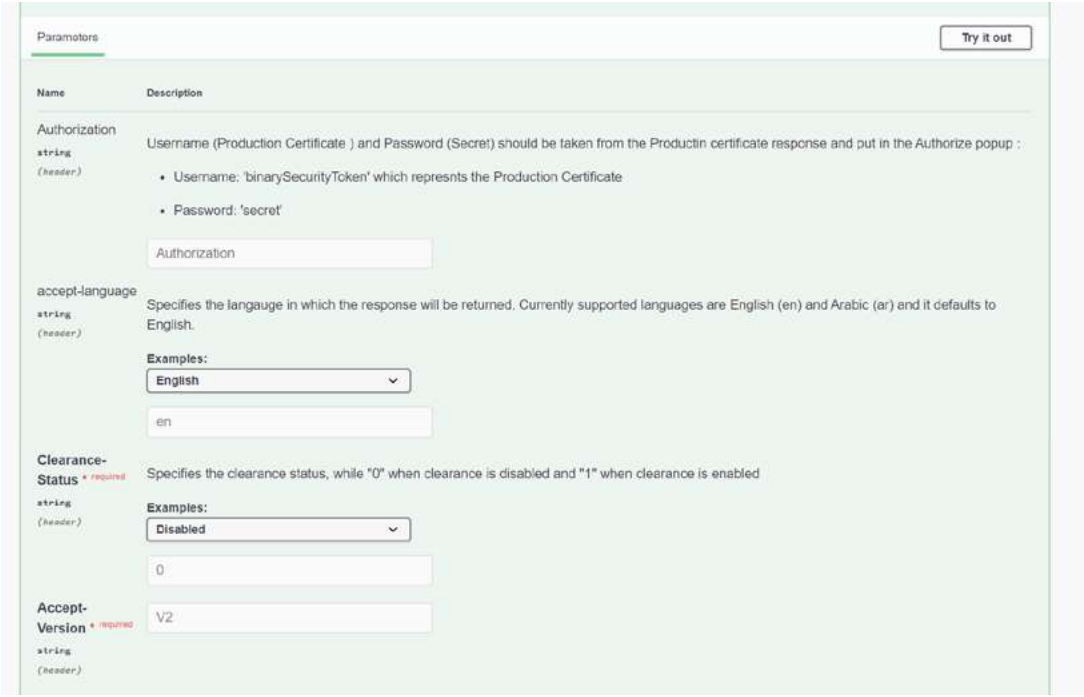
  Please refer to Chapter 3 of this document for further details.



- Step 8: Click on Reporting API drop-down



- Step 9: Click on try it out

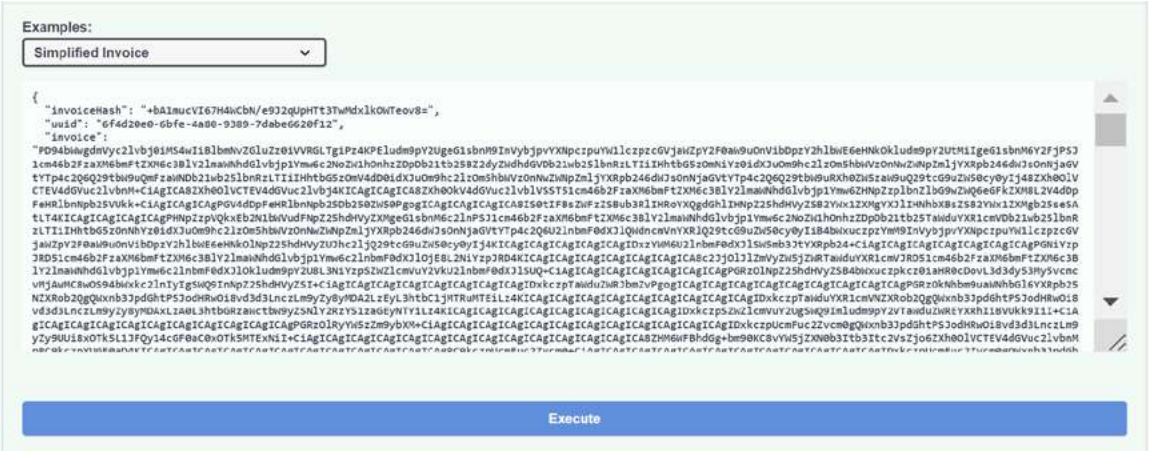


Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

- Step 10: Replace the Invoice hash in xml and encode the xml using Base 64

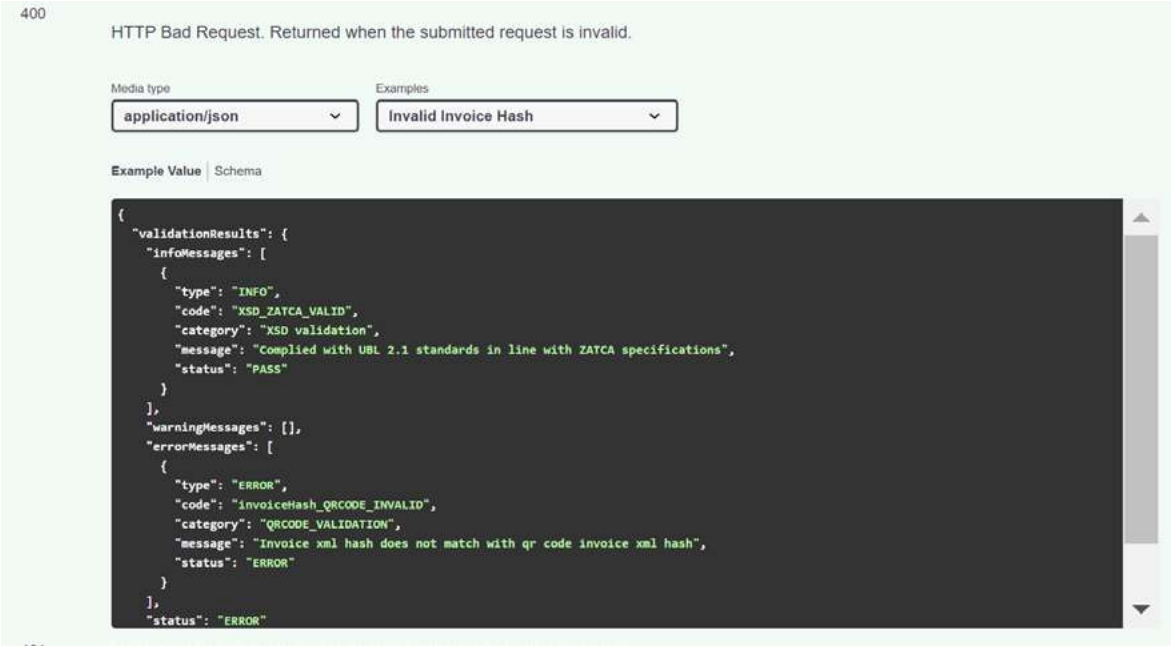
- Step 11:Fill the request body (invoice hash, UUID, Base 64 encoded XML invoice)

- Step 12: Execute



- Result (200)

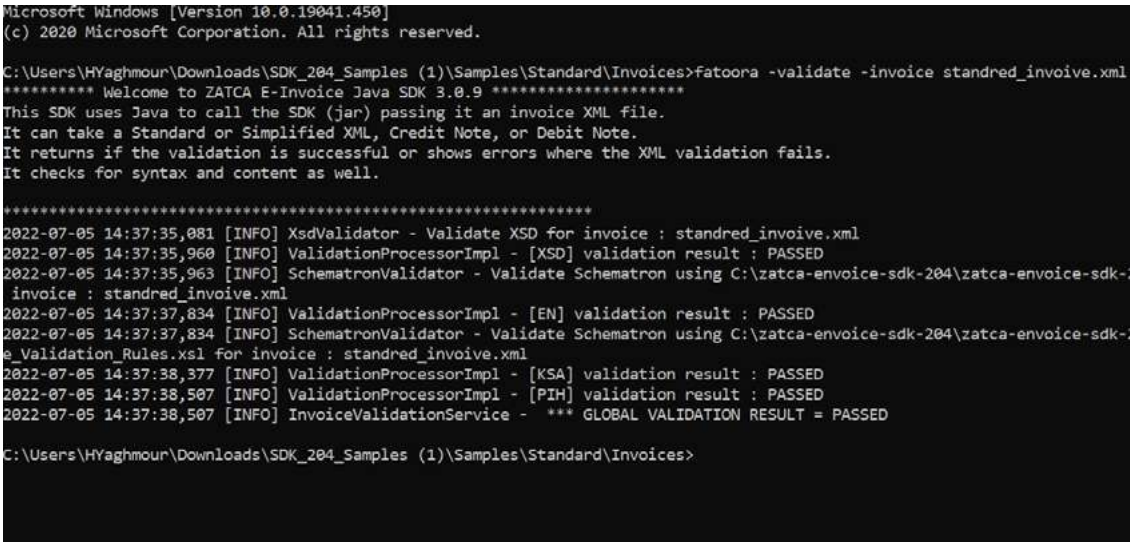
- Result (400)



2.3.10.6 CLEARANCE

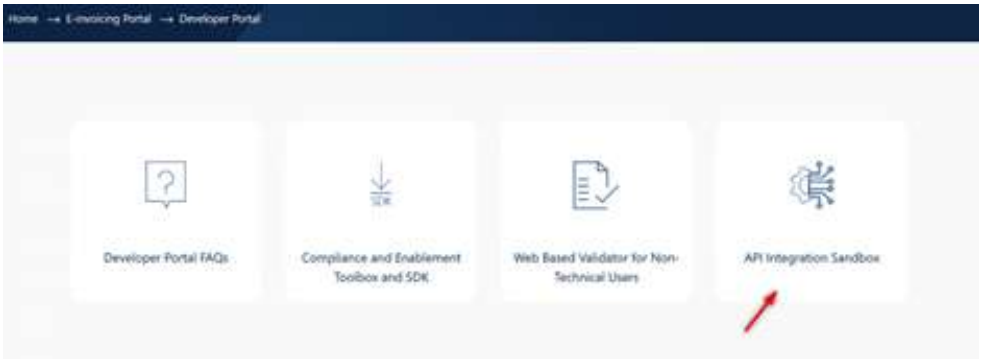
- Step 1: Open CMD and Generate standard invoice

- Step 2: On SDK, get the hash value using the following:
  - fatoora -generateHash -invoice invoiceName.xml
  Afterwards, validate the XML invoice.

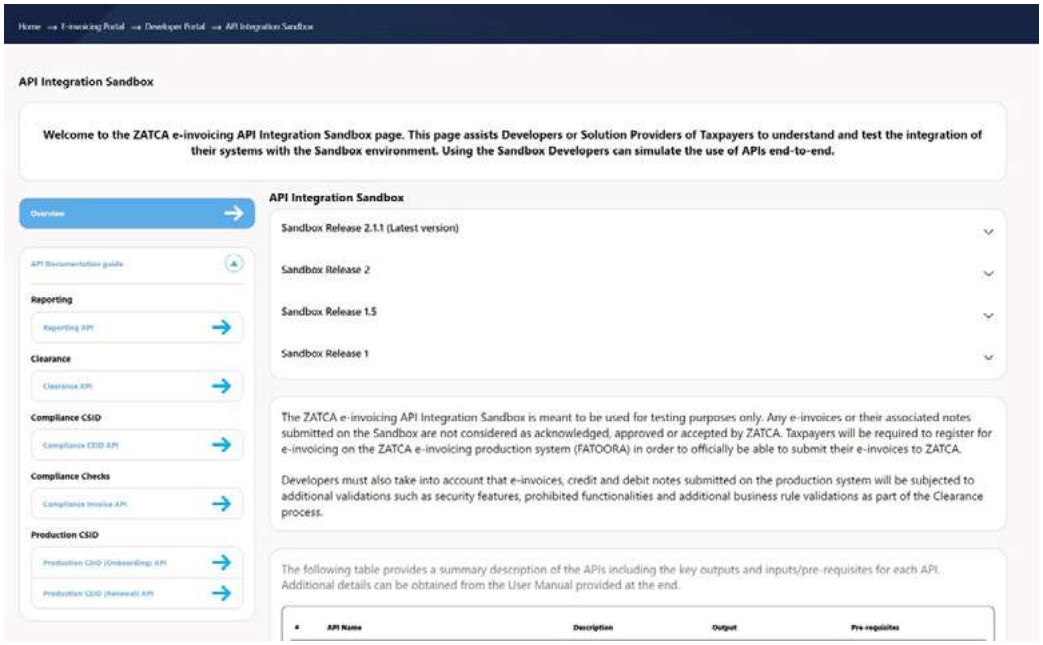


- Step 3: Encode the XML invoice using Base 64

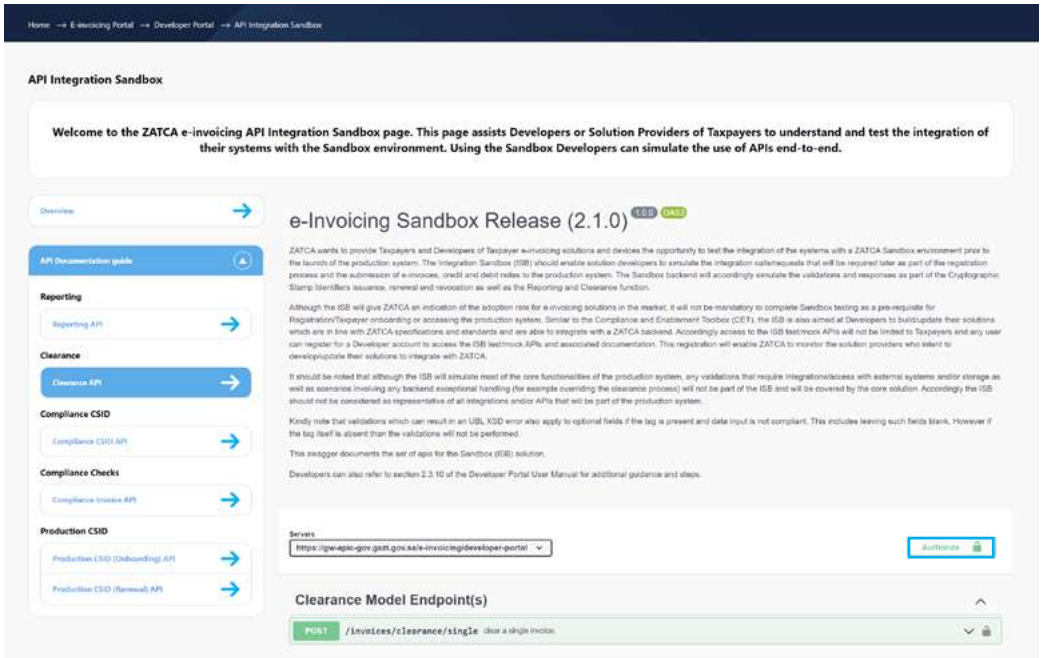
- Step 4: Open Developer Portal and choose integration sandbox



- Step 5: Click on API documentation guides



- Step 6: Click on Authorize Clearance API

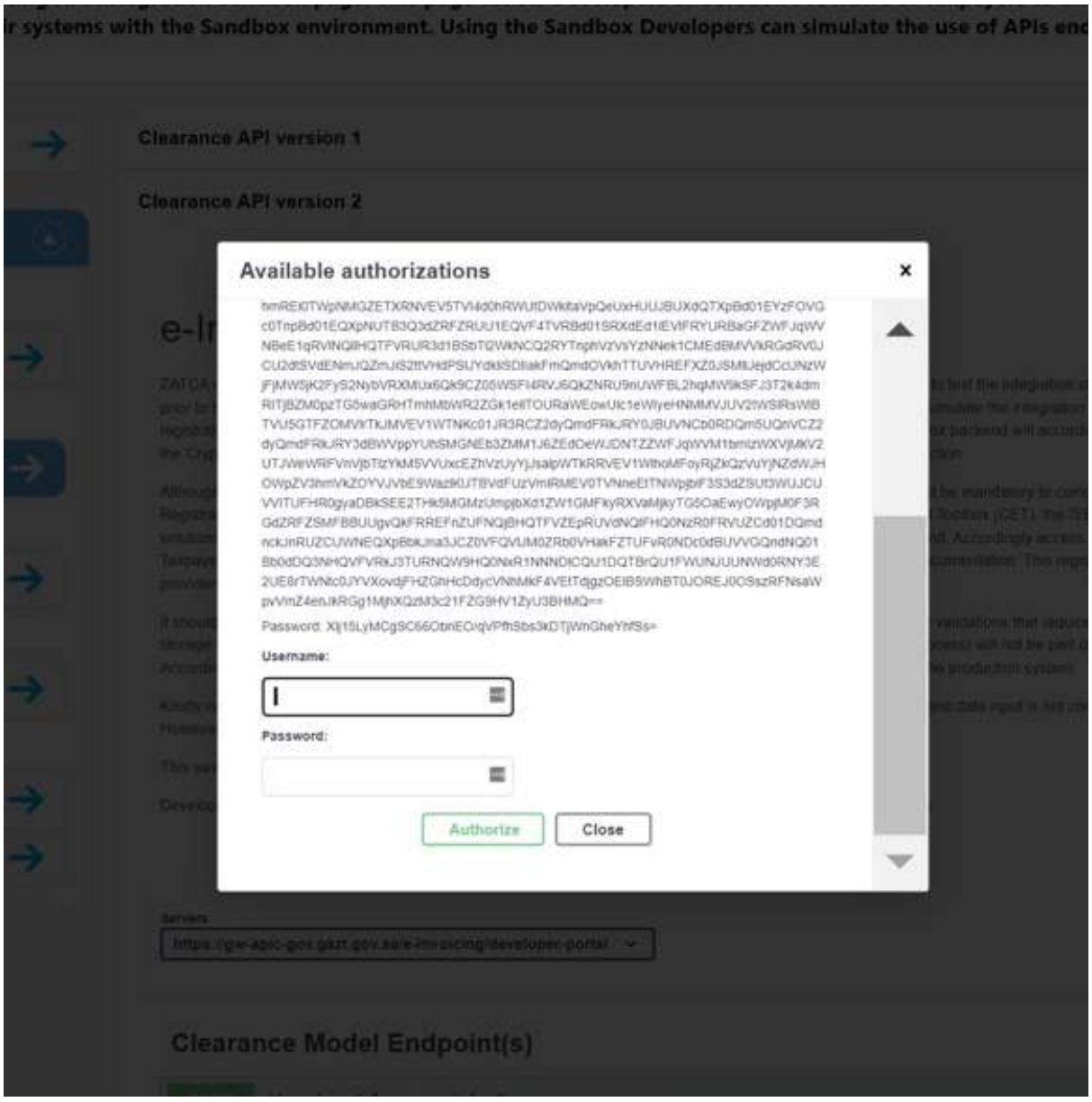


- Step 7:

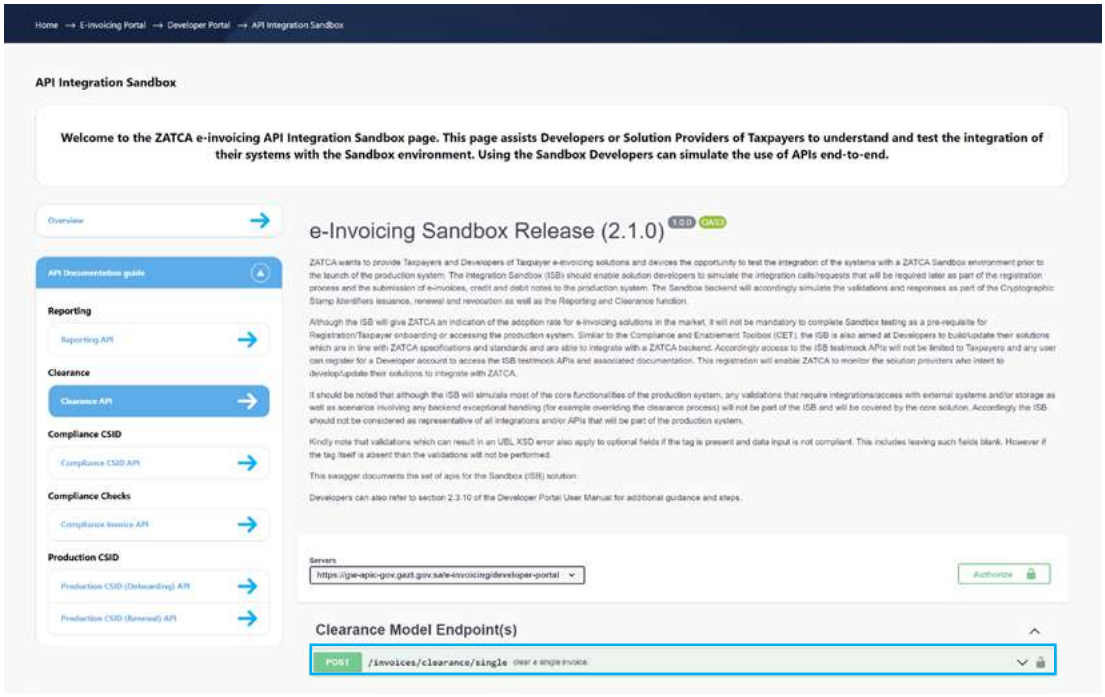
For the Sandbox, use the sample dummy Username and Password provided to you on the Authorization screen.

For Production, run the Compliance CSID API to obtain the "binarySecurityToken" to be used as the Username and "secret" as the Password.

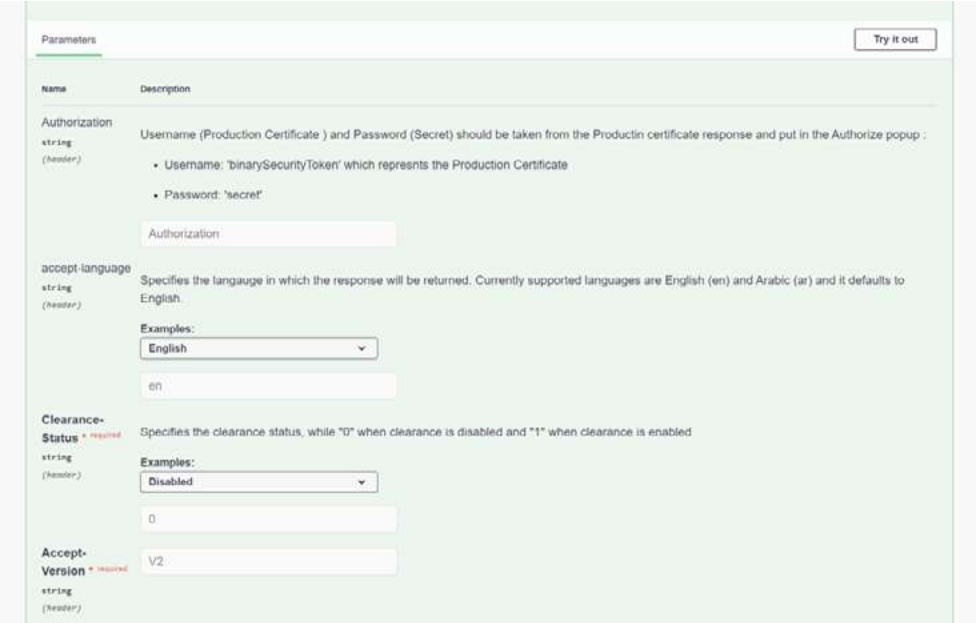
Please refer to Chapter 3 of this document for further details.



- Step 8: Click on Clearance API drop-down

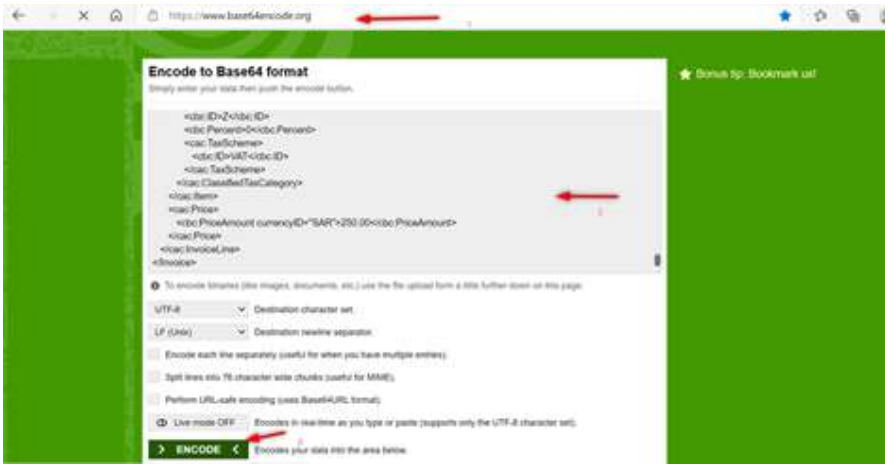


- Step 9: Click on try it out



Note: V2 refers to the Version of the APIs used and should be mentioned in the API calls (V2 is currently the only valid version).

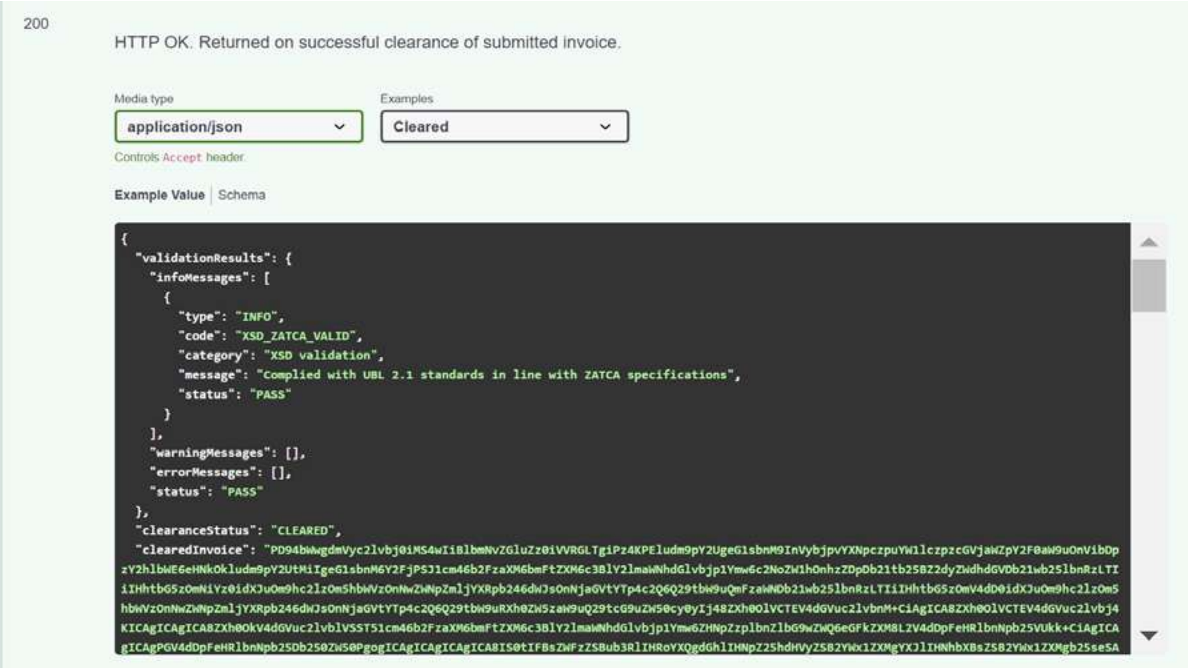
- Step 10: Replace the Invoice hash in xml and encode the xml using Base 64



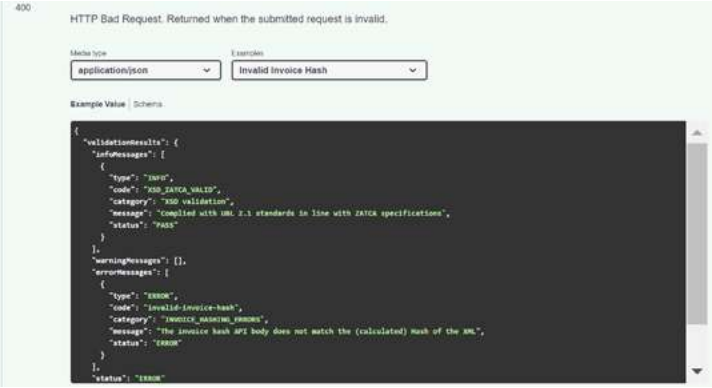
- Step 11: Fill the request body (invoice hash, UUID, Base 64 encoded XML invoice)

- Step 12: Execute

- Result (200)



- Result (400)



2.3.11 API Summary

Table 1

The following table gives a more detailed summary of the differences between the Integration Sandbox releases in terms of the APIs as well associated components. The current release also indicates the new additions/changes made in comparison to the previous release.

Function-ality	Description	Release 1 (No-vember 2021)	Release 1.5 (Feb-ruary 2022)	Release 2 (Current - April 2022)
APIs	The list of APIs that are covered in each release including references to the functionalities they are part of	**Invoices APIs:** • Reporting API • Clearance API **Onboarding APIs:** • CSID API (for • Onboarding) • CSID API (for • Renewal)	**Invoices APIs:** • Reporting API • Clearance API **Onboarding APIs:** • Compliance CSID API • Production CSID API • (for Onboarding) • Production CSID API (for Renewal) • Compliance Checks APIs (for Onboard-ing / Renewal)	**Invoices APIs:** • Reporting API • Clearance API **Onboarding APIs:** • Compliance CSID API • Production CSID API (for On-boarding) • Production CSID API (for Re-newal) • Compliance Checks APIs (for Onboarding / Renewal)

Validation Engine (For Invoices)	The treatment of validations and exceptions as part of the Reporting and Clearance process. Exceptions here refer to warnings which are similar to errors but do not cause the submitted invoices/documents to be rejected but are still indicated in the response so that they can be corrected in future submissions.	• As per original (published) data dictionary, XML Implementation Standards and Security Features and Implementation Standards • No exceptions (Invoices are either accepted or rejected) • Not possible to test for Sandbox behavior When Clearance is disabled	• As per updated data dictionary, XML Implementation Standards and Security Features and Implementation Standards (including updates to CSR and CSID standards) • Seller Address field will be accepted with warning for Taxpayer devices / solution units to differentiate between a warning and an error response • Not possible to test for Sandbox behavior when Clearance is disabled	• As per updated data dictionary, XML Implementation Standards and Security Features and Implementation Standards (including updates to CSR and CSID standards) • Seller Address field will be accepted with warning for Taxpayer devices / solution units to differentiate Between a warning and an error response • Two variants of the Reporting and Clearance APIs which are configured with Clearance disabled is being provided - • NEW • **Note:** In the Core Einvoicing Solution there will only be one API each for Reporting and Clearance which at any point of time willeither be configured to Clearance being enabled or disabled
CSR and CSID (For Onboarding)	The formats and fields for the Certificate Signing Request (CSR) and the resultant Cryptographic Stamp Identifier (CSID) that is used as part of the Onboarding process.	As per the original (published) Security Features and Implementation Standards	As per the updated Security Features and Implementation Standards	As per the updated Security Features and Implementation Standards
Swagger Files (API Specifications)	The API documentation associated with the Swagger files.	• Covers the APIs mentioned above • No provisions for Exceptions or turning off Clearance	• Covers the APIs mentioned • Provision for 1 Exception (Non-compliance in the Seller's Address field is accepted as a warning) • No provisions for turning off Clearance • Covers the two separate APIs for Compliance and Production CSIDs	• Covers the Reporting and Clearance APIs above with provision for 1 Exception • Covers the Reporting and Clearance APIs above with provision for turning off Clearance (through two additional variants of the Reporting and Clearance APIs) - • NEW • Covers the two separate APIs for Compliance and Production CSIDs

Table 2

The following table provides a summary description of the APIs including the key outputs and inputs/pre-requisites for each API.

API Name	Description	Output	Pre-requisites
Reporting API	This API should be used to test submitting Simplified e-invoices, credit or debit note to the ZATCA backend system as part of the Reporting process When Clearance is disabled, this API can also be used to test submitting Standard e-invoices, credit or debit notes for Reporting Note: In the Integration Sandbox there will be two variants of the Reporting API, one which is configured to Clearance being enabled (i.e. it will not accept Standard documents) and one which is configured to Clearance being disabled (i.e. it will also accept Standard documents to be submitted for Reporting)	• **If no errors or warnings: Accepted** • **If error in Seller Address: Accepted with warning message** • **If errors other than Seller Address: Rejected with error messages**	• **A test Production CSID obtained from API #5 or #6 below** • **Simplified invoice, credit or debit note in XML format** • **Standard invoice, credit or debit note in XML format when Clearance is disabled**
Clearance API	This API should be used to test submitting test Standard e-invoices, credit or debit note to the ZATCA backend system as part of the Clearance process When Clearance is disabled, this API will return a 303 Response indicating that the Reporting API be used to submit Standard documents as well Note: In the Integration Sandbox there will be two variants of the Clearance API, one which is configured to Clearance being enabled (i.e. it will validate and clear Standard documents) and one which is configured to Clearance being disabled (i.e. it will return response 303 stating that Clearance is currently disabled and the Reporting API must be used to submit Standard documents as well)	• If no errors or warnings: Accepted and document is returned with test ZATCA stamp and QR code • If error in Seller Address: Accepted with warning message and document is returned with test ZATCA stamp and QR code • If errors other than Seller Address: Rejected with error messages • Response 303 when Clearance is disabled asking the Reporting API to be used to submit Standard documents	• A test Production CSID obtained from API #5 or #6 below • Standard invoice, credit or debit note in XML format

Compliance CSID API	This API should be used to test submitting test CSRs (Certificate Signing Requests) to the ZATCA backend system as part of the Onboarding and renewal process	• Valid request: Test Compliance CSID and a test Request ID are obtained • Invalid request: Error message(s)	• Public Private Key pair • Signed CSR
Production CSID API (for Onboarding)	This API will be used to submit a test Request ID to a test ZATCA backend system as part of the Onboarding process	• Valid request: Test Production CSID is obtained • Invalid request: Error message(s)	• A test Compliance CSID obtained from APIs #3 above • A test (dummy) request ID
Production CSID API (for Renewal)	This API will be used to submit a test Request ID to a test ZATCA backend system as part of the Onboarding process	• Valid request: Test Production CSID is obtained • Invalid request: Error message(s)	• A test Compliance CSID obtained from APIs #3 above • A test (dummy) request ID
Compliance Checks APIs (for Onboarding / Renewal)	These APIs should be used to test the compliance check for the device / solution unit (EGS) as part of the Onboarding and/or Renewal processes The compliance checks include checking compliance of Standard and/or Simplified documents when Clearance is enabled (Compliance Invoice API) or when Clearance is disabled (Compliance Invoice Clearance Disabled API);	• All Compliance checks passed • One or more compliance checks failed with error messages	• A test Compliance CSID obtained from APIs #3 above • Standard and/or Simplified invoices, credit or debit notes in XML format

2.3.12 Accessing the Developer Portal Support Page

The Developer Portal Support Page can be accessed from the main dashboard of the Developer Portal and does not require any prior registration / log in. Through this page, the user can view the different types of support available which includes the Toolbox and Sandbox documentation. In addition, the user can view the FAQ section to find readily available answers to common inquiries they may have on the Developer Portal tools and functionalities as well as more specific questions on testing the compliance of their XMLs. Users can also find the support contact information that they can access should they require any support. This includes phone number / hotline, international phone number and the email address. Users could also provide any suggestions or complaints they may have.

The user can access the Developer Portal Support Page from the main Developer Portal page.

The following categories are available to users:

- General support
- SDK support
- Integration Sandbox support
- Compliance and Enablement Toolbox support

A search bar is also readily available for users to search and obtain the relevant information easily. The user can view common enquiries in the FAQ page. The user can see a contact section at the bottom of the support page in case of experiencing any issues and in the event that the user would want to receive the support of the contact center.