

IRAS API SERVICES INTERFACE SPECIFICATIONS

Check GST Register

Last updated on : 30 May 2022

Version No: 1.0.6

Disclaimers: The information provided is intended for better general understanding and is not intended to comprehensively address all possible issues that may arise. The contents are provided on an “as is” basis without warranties of any kind. IRAS shall not be liable for any damages, expenses, costs or loss of any kind however caused as a result of, or in connection with your use of this document. While every effort has been made to ensure that the above information is consistent with existing policies and practice, should there be any changes, IRAS reserves the right to vary our position accordingly.

© Inland Revenue Authority of Singapore

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, including photocopying and recording without the written permission of the copyright holder, application for which should be addressed to the publisher. Such written permission must also be obtained before any part of this publication is stored in a retrieval system of any nature.

Version Control

Version History

Revision Date	Version Number	Change Log Summary
30 May 2022	1.0.6	Added Remark in Response Payload

Table of Contents

1.	Introduction	4
2.	Registration	4
3.	API Services	5
3.1	Network Security	5
3.2	Message Security	6
3.3	General Usage	6
3.4	Sandbox Usage	6
3.5	Production Usage	7
3.6	Common Interface Information	7
3.7	Common Response Payload	8
4.	Check GST Register API	9
4.1	API Endpoint	9
4.2	Request Payload	9
4.3	Response Payload	9
4.4	Sandbox Testing	11
4.4.1	GST Registration Search Service	11
5.	Sample Code (C#)	13
6.	List of Possible Error Codes	14

1. Introduction

The Inland Revenue Authority of Singapore (IRAS) provides application programming interface (API) services to allow application developers to submit and retrieve tax related matters using HTTP requests. Most of the APIs will be in the form of a JSON web service which reduces client/server coupling and thus enabling easier integration between IRAS' services with external developers.

While some services require a simple GET, others may be secured and require credentials that can be passed via HTTP header parameters which are as follows :

X-IBM-Client-Id	String containing the client ID of the application invoking IRAS API. This value will be provided to the application vendor by IRAS. E.g. a1234b5c-1234-abcd-efgh-a1234b5cdef
X-IBM-Client-Secret	String containing the client secret of the application invoking IRAS API. This value will be provided to the application vendor by IRAS. E.g. a12345bC67e8fG9a12345bC67e8fG9a12345bC67e8fG9

This document serves as a guide for developers to consume IRAS API services.

2. Registration

Developers are required to test APIs in the Sandbox environment first, before using them in Production. Create a developer account at <https://apisandbox.iras.gov.sg/> to subscribe to IRAS APIs for Sandbox Testing, and a developer account at <https://apiservices.iras.gov.sg/> to subscribe to IRAS APIs for Production use.

A computer-generated email will be sent to the subscriber's email account for account activation of the API Marketplace.

Refer to <https://go.gov.sg/iras-apiuserguide> for more details on how to get started.

3. API Services

The following sections describe the request and response for each of the API services.

The table below shows the list of Check GST Register API services currently available in IRAS.

Name of API Service	Description	Endpoint
Check GST Register API	The Check GST Register API enables you to check whether businesses are GST-registered based on their GST registration number, UEN or NRIC.	For Sandbox Testing: https://apisandbox.iras.gov.sg/iras/sb/GSTListing/SearchGSTRegistered
		For Production Usage: https://apiservices.iras.gov.sg/iras/prod/GSTListing/SearchGSTRegistered

3.1 Network Security

The API endpoint has to be triggered from a Server-to-Server connection. Transport Layer Security (TLS) and IP Whitelisting will be used to secure end-to-end communications between the server invoking the API and IRAS API Gateway.

Your server, software or application must be able to:

1. Support the following protocols: HTTP/2, TLS 1.2/1.3, and
2. Trigger HTTP/GET and HTTP/POST requests.

3.2 Message Security

The following parameters must be populated in the HTTP header:

X-IBM-Client-Id	String containing the client ID of the application invoking IRAS API. This value will be provided to the application vendor by IRAS.
X-IBM-Client-Secret	String containing the client secret of the application invoking IRAS API. This value will be provided to the application vendor by IRAS.
Content-Type	application/json

Sample HTTP Header in a HTTP Post Request

```
POST /iras/prod/eStamp/SalePurchaseBuyers HTTP/1.1
Host: https://apiservices.iras.gov.sg
X-IBM-Client-Id: 40e7be2f-0b4f-4985-bcc9-cdfd38c5b5c8
X-IBM-Client-Secret: vD0kR8iT3kR1hB8dPlqS3hC4lJ1aA1fV4pQ0uW0hI3uI5bW4rU
Content-Type: application/json
Accept: application/json
```

3.3 General Usage

Each API request payload is limited to 2 MB in total data size.

3.4 Sandbox Usage

Approval **is required** to use this service in the Sandbox environment.

API services in the Sandbox environment are designed to mimic the Production environment so that developers can perform integration tests on the API before consuming actual data from the Production environment.

You are required to perform Sandbox Testing before you can use the API in Production. Refer to the Sandbox Testing section below for a list of tests to perform in Sandbox

3.5 Production Usage

Approval **is required** to use this service in the Production environment.

Subscribe to the API Plan to request for approval. An email notification will be sent to your registered email when your request has been processed. If approval is granted by IRAS, the client application can successfully consume the API.

Use of IRAS API(s) are governed by the [Additional API Terms of Use](#) found at the API Marketplace.

The use of IRAS API(s) in the Production environment is only intended for the purpose as indicated in the API service description. Any other forms of use, **including all forms of tests**, are strictly prohibited. In the event such activities are found to have occurred in the Production environment, you will be deemed to have breached the API Terms of Use, which may result in the suspension or termination of your account.

3.6 Common Interface Information

- JSON is case sensitive by specifications.
- All date strings are to be represented in compliance to the [ISO-8601](#) standard.
- All string fields are subject to validation of the following acceptable characters that is allowed (in red):
 - [a-zA-Z0-9'@#()~./&+ _] (**Note:** whitespace is included)
- All properties follow the camel-case convention.
- Unless stated as optional, all JSON object properties must be specified.
- Unless otherwise specified, all JSON services are invoked using HTTP verb POST.
- All input data format are as specified like the following:

Data Type and Size	Description	Example
String(12)	A string containing maximum 12 characters.	"180084010K"
Number(4)	A numeric value containing maximum 4 whole numbers.	1990

3.7 Common Response Payload

All response payloads share the following common fields:

Parameter Name	Data Type	Description
data	Object	The data property will be populated differently based on the API that is being invoked.
returnCode	Integer	10 : Success - The request was successfully processed. 20 : Warning - The request was successfully processed. However, there are non-fatal issues. Please refer to the “info” object for diagnostic information. 30 : Failure – The request was not processed. Refer to “info” object for error information.
info	Object	This complex object holds any diagnostic information that will allow developers to debug their failed requests.
info.message	String	Diagnostic message in the event of warning or error.
Info.messageCode	Integer	Integer code signifying the type of error or warning. 850300 : Request object is null – The incoming JSON request is null. 850301 : Arguments error – There is an error with one of the arguments provided. 850302 : Generic error – There is an exception within the service. 850303 : Service is inactive. 850304 : Service is not authorized for usage based on the provided credentials. 850305 : Invalid test user – The input fields provided are not valid for sandbox testing. 850806 : Invalid document reference number – The provided document reference number is not found in our system. 850807 : DRN does not match contracting party's ID – The provided document reference number does not match the contracting party's identity number. 400033 : No match data found
info.fieldInfoList	Array	An array for FieldInfo objects.
info.fieldInfoList.field	String	Name of the field that resulted in a warning / error.
Info.fieldInfoList.message	String	Diagnostic message provided to aid consumer's developers.

4. Check GST Register API

4.1 API Endpoint

Method	Endpoint
POST	/GSTListing/SearchGSTRegistered

4.2 Request Payload

Parameter Name	Data Type (Max Length)	Mandatory / Optional / Conditional	Description
clientID	String	Optional	This has to match the client id that is passed in via the HTTP headers
regID	String	Mandatory	The ID of the GST-registered entity (Search by Unique Entity Number/ GST Registration Number/ NRIC)

Request Payload Sample

```
{
  "clientID": "YOUR_CLIENT_ID",
  "regID": "200312345A"
}
```

4.3 Response Payload

Parameter Name	Data Type	Description
data	Object	The object payload containing information about the GST registered entity.
data.name	String	The name of the organisation or company. (Note: will not be provided if the regID is NRIC or GST Registration Number of individual sole-proprietor)
data.gstRegistrationNumber	String	The GST Registration Number.
data.registrationId	Number	The ID of the GST-registered entity. (Note: will not be provided if the regID is a GST Registration Number of individual sole-proprietor)
data.RegisteredFrom	String	Date of registration, if any. Represented in ISO-8601 string.

© Inland Revenue Authority of Singapore

No part of this publication may be reproduced or transmitted in any form or by any means, including photocopying and recording without the written permission of the copyright holder, application for which should be addressed to the publisher.

data.RegisteredTo	String	Last Day of registration, if any. Represented in ISO-8601 string.
data.Status	String	Status of GST Registration.
data.Remark	String	Additional remarks, if any. If no remark, to show dash "-".
returnCode	As per Section 3.7	
info		
info.message		
Info.messageCode		
info.fieldInfoList		
info.fieldInfoList.field		
Info.fieldInfoList.message		

Successful response payload sample

```
{
  "returnCode": 10,
  "data": {
    "gstRegistrationNumber": "200312345A",
    "name": "MY_COMPANY_NAME",
    "registrationId": "200312345A",
    "RegisteredFrom": "1994-04-01T00:00:00",
    "Status": "Registered",
    "Remark": "Currently registered under Simplified Pay-Only Regime"
  },
  "info": {
    "fieldInfoList": []
  }
}
```

Error response payload sample

```
{
  "returnCode": 30,
  "info": {
    "fieldInfoList": [
      {
        "field": "regId",
        "message": "Value is not valid"
      }
    ],
    "message": "Arguments Error",
    "messageCode": 850301
  }
}
```

© Inland Revenue Authority of Singapore

No part of this publication may be reproduced or transmitted in any form or by any means, including photocopying and recording without the written permission of the copyright holder, application for which should be addressed to the publisher.

4.4 Sandbox Testing

As explained in [Section 2](#), developers can first create an account in the Sandbox environment to make API calls to our Sandbox URL. This allows the developers to mimic the characteristics of our production environment and create a simulated response from our API.

4.4.1 GST Registration Search Service

<u>Input</u>	<u>Expected Output</u>
<pre>{ "clientID": "YOUR_CLIENT_ID", "regID": "T9100001B" }</pre>	<pre>{ "returnCode": 10, "data": { "gstRegistrationNumber": "M99600001J", "registrationId": "T9100001B", "RegisteredFrom": "2014-03-14T00:00:00", "Status": "Registered", "Remark": " Currently registered under Simplified Pay-Only Regime" }, "info": { "fieldInfoList": [] } }</pre>
<pre>{ "clientID": "YOUR_CLIENT_ID", "regID": " M99600001J" }</pre>	<pre>{ "returnCode": 10, "data": { "gstRegistrationNumber": "M99600001J", "RegisteredFrom": "2014-03-14T00:00:00", "Status": "Registered", "Remark": "-" }, "info": { "fieldInfoList": [] } }</pre>
<pre>{ "clientID": "YOUR_CLIENT_ID", "regID": "95000002K" }</pre>	<pre>{ "returnCode": 10, "data": { "gstRegistrationNumber": "95000002K", "name": "XXX", "registrationId": "95000002K", "RegisteredFrom": "1994-04-01T00:00:00", "RegisteredTo": "1999-06-30T00:00:00", "Status": "Deregistered", "Remark": "-" }, "info": { "fieldInfoList": [] } }</pre>

<pre>{ "clientID": "YOUR_CLIENT_ID", "regID": "208000002D" }</pre>	<pre>{ "returnCode": 10, "data": { "gstRegistrationNumber": "208000002D", "name": "TEST_SANDBOX_LIMO_SERVICE", "registrationId": "208000002D", "RegisteredFrom": "1994-04-01T00:00:00", "Status": "Registered", "Remark": "-" }, "info": { "fieldInfoList": [] } }</pre>
--	--

5. Sample Code (C#)

```

using System;
using System.Net;
using System.IO;
using System.Text;

// jsonData – contains data from Section 3.1.1 of this document
public static void callWebAPI(string jsonData, string url)
{
    try
    {
        var httpWebRequest = (HttpWebRequest)WebRequest.Create(url);
        httpWebRequest.ContentType = "application/json";
        httpWebRequest.Method = "POST";

        //Step 1: Enter the Client-Id given by IRAS
        httpWebRequest.Headers["X-IBM-Client-Id"] = "{YOUR_CLIENT_ID}";
        //Step 2: Enter the Client-Secret given by IRAS
        httpWebRequest.Headers["X-IBM-Client-Secret"] = "{YOUR_CLIENT_SECRET}";

        // Step 3: Call API using POST
        using (var streamWriter = new StreamWriter(httpWebRequest.GetRequestStream()))
        {
            streamWriter.Write(jsonData);
            streamWriter.Flush();
            streamWriter.Close();
        }

        // Step 3a: Output response
        var httpResponse = (HttpWebResponse)httpWebRequest.GetResponse();
        using (var streamReader = new StreamReader(httpResponse.GetResponseStream()))
        {
            var result = streamReader.ReadToEnd();
            //print the received response
            Console.WriteLine(result);
        }
    }
    catch (WebException e)
    {
        if (!string.IsNullOrEmpty(e.Message))
        {
            // Step 3b: Print general errors
            Console.WriteLine("Exception - ");
            Console.WriteLine(e.Message);
        }

        if (e.Response != null)
        {
            // Step 3c: Print Output response exception
            Stream receiveStream = e.Response.GetResponseStream();
            StreamReader readStream = new StreamReader(receiveStream, Encoding.UTF8);
            // print the error received from Server
            Console.WriteLine("Response error received - ");
            Console.WriteLine(readStream.ReadToEnd());
        }
    }
}

```

6. List of Possible Error Codes

The statusCode field will always contain an integer representing the processed state of the request. The list of possible status codes and what they represent are listed below.

HTTP Code	HTTP Message	Possible error scenario and resolution
200	STATUS_OK	The request completed successfully.
204	STATUS_NO_CONTENT	The server has fulfilled the request, but there is no new information to send back.
301	STATUS_MOVED	The requested service has been assigned to a new permanent Uniform Resource Identifier (URI), and any future references to this service should be done using one of the returned URIs.
302	STATUS_REDIRECT	The requested service resides temporarily under a different URI.
304	STATUS_NOT_MODIFIED	The requested did not make any modification.
400	STATUS_BAD_REQUEST	The request could not be processed by the server due to invalid inputs.
401	STATUS_DENIED	The requested service requires user authentication/authorisation.
404	STATUS_NOT_FOUND	The requested resource cannot be found.
410	STATUS_GONE	The requested service is no longer available at the server, and no forwarding address is known.
413	STATUS_REQUEST_TOO_LARGE	The server cannot process the request because the submitted entity is larger than the server is able to process.
429	STATUS_TOO_MANY_REQUESTS	API Rate Limit exceeded for your subscription plan.
500	STATUS_SERVER_ERROR	The server encountered an unexpected condition that prevented it from fulfilling the request.