MANUAL DE SERVICIOS WEB

PLATAFORMA NUEVA GRE

INDICE

1. GENERALIDADES....3
1.1. Tipo de servicio....3
1.2. Autenticación....3
1.3. Manejo de errores....7
ANEXOS....9
I. Relación de errores generales....9

1. GENERALIDADES

1.1. Tipo de servicio

Los servicios web descritos en el presente manual son de tipo REST.

Las URI colocadas en cada servicio son referenciales.

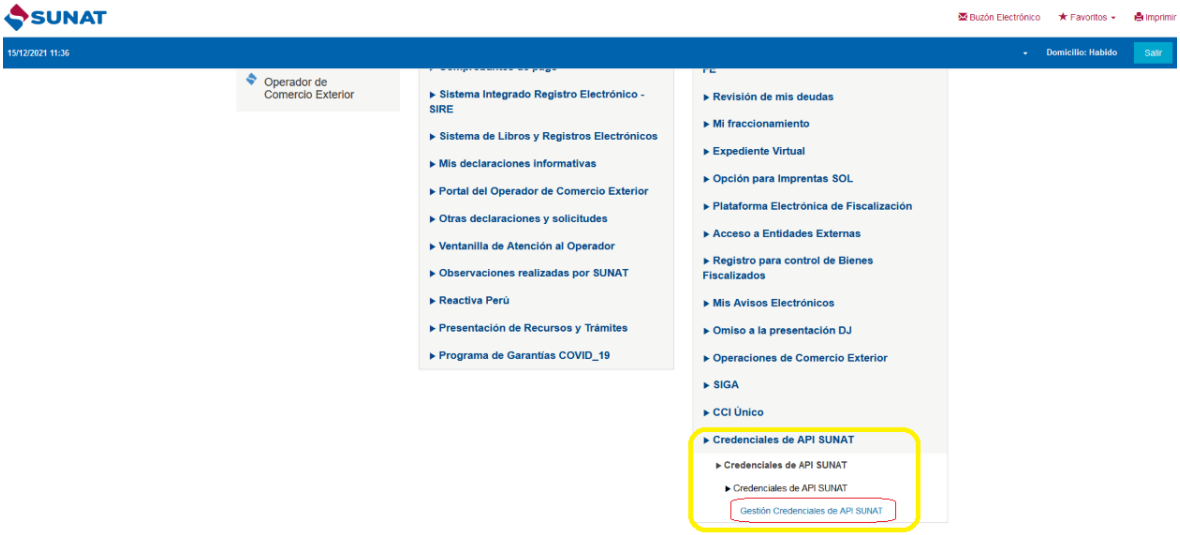
1.2. Autenticación

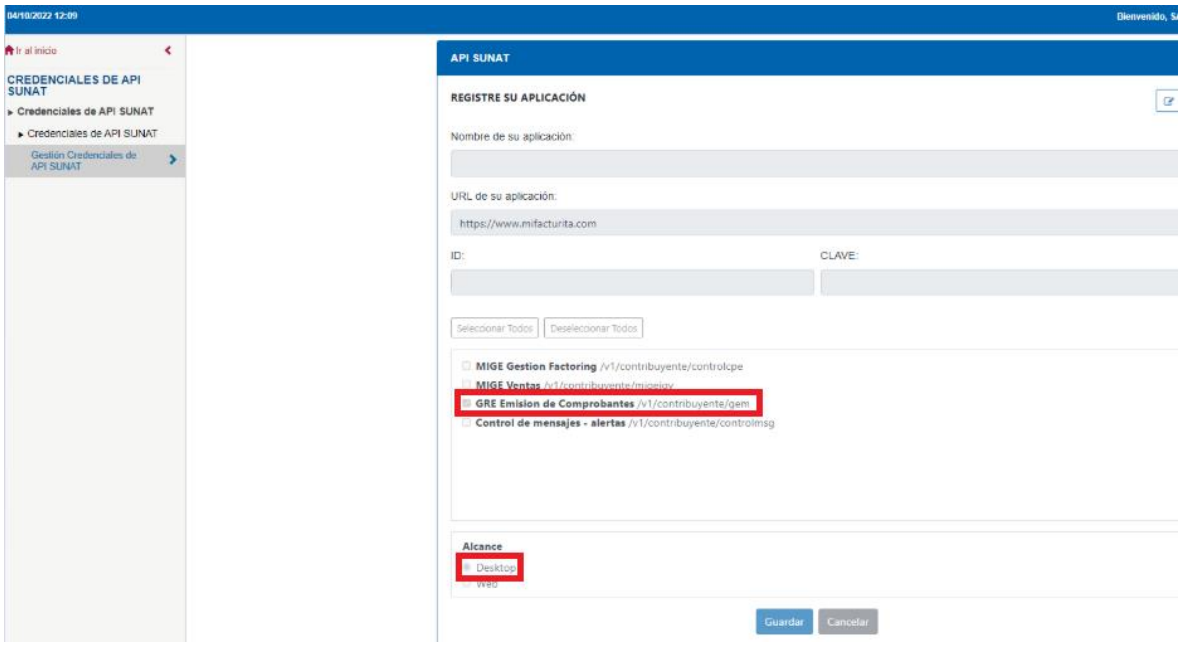
Los servicios web utilizan la autenticación basada en tokens. Para hacer uso de los servicios, el usuario debe seguir los siguientes pasos:

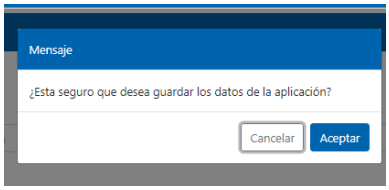
a) Generación de credenciales

En el menú SOL, debe inscribir la aplicación que usará los servicios REST y generar sus credenciales (client_id y client_secret). Este paso se realizará por única vez.

La ubicación de la opción en el menú sol es la siguiente: Credenciales de API SUNAT/ Credenciales de API SUNAT/ Credenciales de API SUNAT/ Credenciales de API SUNAT.







b) Generación del token

Con las credenciales generadas en el punto anterior, generará un token a través de un servicio que pondrá disponible SUNAT para tal fin. Este token tendrá una vigencia la cual se indica en el retorno del servicio (actualmente es de 1 hora) y dentro de este periodo, podrá utilizarse las veces que requiera invocar los servicios.

Se deberá acceder a la siguiente URL, como "POST":
https://api-seguridad.sunat.gob.pe/v1/clientessol/<client_id>/oauth2/token/

Donde:

- **<client_id>**: Es el client_id generado en menú SOL
- La URI colocada es referencial

En la cabecera (Header) se debe enviar el siguiente parámetro:

Content-Type	Application/json

El cuerpo (Body) de la consulta deberá ser de tipo "x-www-form-urlencoded" y se debe enviar los siguientes parámetros:

grant_type	password

scope	https://api-cpe.sunat.gob.pe
client_id	<client_id> generado en menú SOL
client_secret	<client_secret> generado en menú SOL
username	<Número de RUC> + <Usuario SOL>
password	<Contraseña SOL>

Donde:

- El parámetro grant_type tiene valor fijo

Y como datos de respuesta se tendrá:

access_token	(token generado)
token_type	(tipo de token)
expires_in	(tiempo de expiración del token - El tiempo de expiración es en segundos. Concluido el tiempo podrá generar un nuevo token)

A continuación, se muestra un ejemplo del JSON para la generación de token usando la herramienta REST "POSTMAN"[1] :

```
{
  "name": "Token Password",
  "request": {
    "method": "POST",
    "header": [
      {
        "key": "Content-Type",
        "name": "Content-Type",
        "type": "text",
        "value": "application/json"
      }
    ],
    "body": {
      "mode": "urlencoded",
      "urlencoded": [
        {
          "key": "grant_type",
          "value": "password",
          "type": "text"
        },
        {
          "key": "client_id",
          "value": "e73c9bdb-8a64-4e3e-ab6e-e4f4d71b6570",
          "type": "text"
        },
        {
          "key": "client_secret",
          "value": "i2FX+wC96UQaCQXTb0p3Blv7G0WaLiHk1HhAa4XVYb",
          "type": "text"
        },
        {
          "key": "username",
          "value": "20552174918HYSJU872",
          "type": "text"
        },
        {
          "key": "password",
          "value": "YUSIW873",
          "type": "text"
        }
      ]
    },
    "url": {
      "raw": "https://api-seguridad-test.sunat.gob.pe:444/v1/clientessol/e73c9bdb-8a64-4e3e-ab6e-e4f4d71b6570/oauth2/token/",
      "protocol": "https",
      "host": [
        "api-seguridad-test",
        "sunat",
        "gob",
        "pe"
      ],
      "port": "444",
      "path": [
        "v1",
        "clientessol",
        "e73c9bdb-8a64-4e3e-ab6e-e4f4d71b6570",
        "oauth2",
        "token",
        ""
      ]
    }
  },
  "response": []
}
```


(*) Los datos de la imagen son referenciales

[1] Disponible para los sistemas operativos: Windows, Mac y Linux.

Respuesta de la consulta:

c) Uso del token

Para el uso de los servicios, se deberá haber generado previamente el token en la sección anterior. El token se usará de la siguiente forma en la invocación del servicio:

En las Cabeceras (Headers) se deberá enviar lo siguiente:

Authorization	Bearer + token

Se debe enviar la palabra "Bearer" concatenado con un espacio y luego el token generado.

1.3. Manejo de errores

Se tienen dos niveles de verificación, uno general relacionado a la invocación y conectividad y es común a todos los servicios, y un segundo nivel específico acorde a las características propias del servicio que se está usando.

A) Nivel general

Cuando se presenta un error de tipo general, el servicio responde con los siguientes parámetros:

Response Header	
Parámetros	**Valor**
HTTP status	Código de Error HTTP
Content-Type	application/json

Response Body		
Parámetros de Salida	**Descripcion**	**Tipo dato**
cod	Código de error	String
msg	Mensaje de error para el usuario	String
exc	Traza del error	String

Ejemplo de Response

```
{
   "cod":"500",
   "msg":"Internal Server Error - Se presento una condicion inesperada que impidio completar el Request",
   "exc":"java.lang.NullPointerException at ..."
}
```

La relación de los principales errores generales se encuentra en el Anexo I.

B) Nivel específico

Si se trata de un error propio del servicio que se está invocando, el sistema retornará un error similar al anterior con el código de error HTTP igual a 422. Adicionalmente, mostrará el código de error específico de las validaciones funcionales del servicio.

Response Body		
Parámetros de Salida	**Descripcion**	**Tipo dato**
cod	Código de error (Mostrará 422)	String
msg	Mensaje de error para el usuario	String
exc	Traza del error	String
errors	Array de errores y descripción del error	String

Ejemplo de Response


```
{
   "cod": "422",
   "msg": "Unprocessable Entity - Se presentaron errores de validacion que impidieron completar el Request",
   "exc": null,
   "errors": [
      {
         "codError": "166",
         "desError": "Código de ticket no enviado."
      }
   ]
}
```

ANEXOS

I. Relación de errores generales

Código de error	Descripción del mensaje de error
400	Bad Request - El Request no puede ser entendido por el Servidor debido a errores de Sintaxis, El cliente no debe repetir el Request sin modificaciones
401	Unauthorized - Fallo en la autenticación del Cliente
403	Forbidden - El Cliente no tiene autorización para acceder al Recurso
404	Not Found - El Recurso Solicitado no puede ser encontrado
405	Not Allowed - El Método HTTP utilizado en el Request no es soportado por el Recurso
406	Not Acceptable - El Recurso no puede responder al Cliente en el Media Type solicitado en el Request
415	Unsupported Media Type - La Entidad en el Body del Request está en un Media Type que no es soportado por el Recurso
422	Unprocessable Entity - Se presentaron errores de validacion que impidieron completar el Request
500	Internal Server Error - Se presento una condición inesperada que impidió completar el Request
503	Service Unavailable - El Servidor no está disponible temporalmente o está muy ocupado para responder al Request