



**CLEVERSOFT**

SISTEMAS DE GESTIÓN EMPRESARIAL

# **Manual de la API (application programming interface)**

## Contenido

Activación Licencias para Clientes .....	3
Instalación de la API .....	5
Uso de la página Test de la API .....	8
Configuración de las Dirección WEB para el cliente .....	9
Alta de user del cliente .....	11
Configurar método de pago .....	13
Creación de una función .....	14
Como insertar el AutoDoc (Documentación).....	16



## Activación Licencias para Clientes

Para que un cliente pueda hacer uso de los servicios de API de Cleversoft debe estar correctamente configurado en el administrador de licencias, para ello seguir los siguientes pasos:

**PASO 1:** Ir a la página de licencias (<https://cleversoftsolutions.com/lic/login.php>), Username: Consulta



Control de Licencias CleverSoft  
Login

Username:  
consulta

Password:  
.....

Recordar Ingreso

Ingresar

E ir a la pestaña Clientes



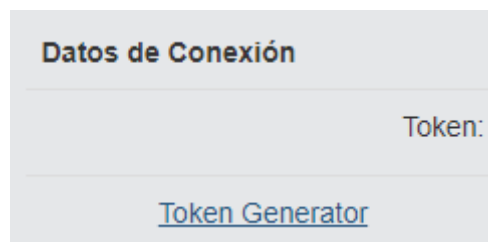
**PASO 2:** Ubicar el cliente y presionar editar

Accion	Ver	Editar	Cliente
Eliminar	Ver	Editar	381

**PASO 3:** Generar Token:

Para ello vamos al link <http://tools.andiditrich.de/Token.html>

O presionamos el link en token generator



Datos de Conexión

Token:

[Token Generator](#)

En la página seleccionar T32 – (AlphaNum Chars)

Base64

Token Generator

Markdown Renderer

# Token Generator

Generate random Tokens

Generate ➤

Generator Mode

T32 - (AlphaNum Chars)

Presionar Generate y copiar alguno de los que se generaron.

## Tokens

```
HbaUuxtJxLCJzyUHMamofMJS5do82Tez  
cMkrCol2PH5n0rSYmySxe3yKFlg4Z1mC  
94e7cs4zqwyECySouZKLge5wD3IlI2k8  
MnF1hIKVy9YyMm41PI4FR0yMPKw8zvr3  
ifpeipcxUpJplfrZOrIHvKntqSeByMHK  
E3W5kLj09FzQMc7NuWVb98VjYtSxS2pJ  
u9fId8pQNXG2nUdlWkBZizv7FIwQInWN  
cosTwIcKn0dZ8P33pamoPx4wSLFTJbMY  
YyI3gtr0cIGy0sbINq0GcPYqzbzbfyT0z  
c0RpsJbDo9Mdx1NxVJ4e2IUx3B6q15UD
```

**PASO 4:** Configurar los Datos de Conexión, colocar Token y URL API Local

**URL API Local:** Es la dirección donde está instalada la API en el server del cliente.

**Token:** este debe ser generado (Paso 3), copiado y pegado en el campo

Datos de Conexión	
Token: <input type="text" value="yFi70xAFi08QspPWFll03JbrBZC5vww"/>	URL Api Local: <input type="text" value="http://192.168.1.12:10680/api2/"/>
<input type="button" value="Test Kiosko"/>	<input type="button" value="Test Api"/>

**PASO 5:** Guardar los cambios, introduciendo User, Clave y presionando la tecla guardar.

User: <input type="text"/>	Clave: <input type="text"/>	<input type="button" value="Guardar"/>
----------------------------	-----------------------------	--

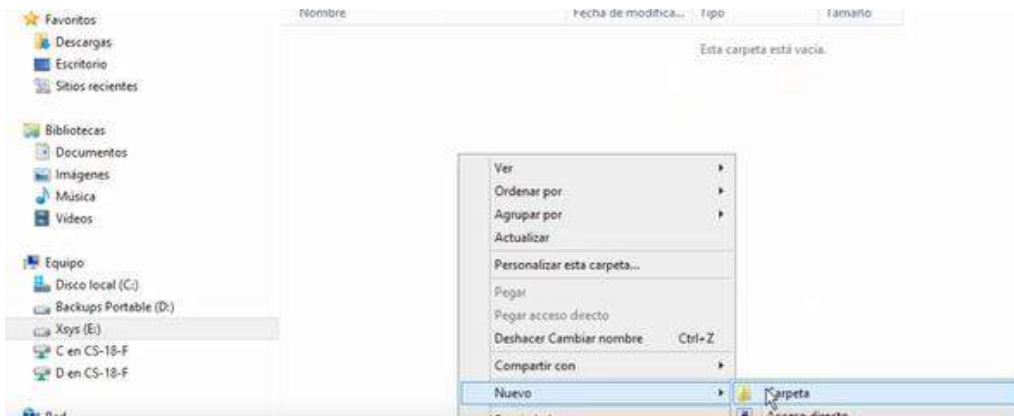
## Instalación de la API

Para que la aplicación funcione debe estar instalado el servidor apache (Xampp), para instalar y configurar seguir instructivo del siguiente link <https://cleversoftsolutions.com/soporte/kb/faq.php?id=325>

Para realizar la instalación de la API, se deben seguir los siguientes pasos:

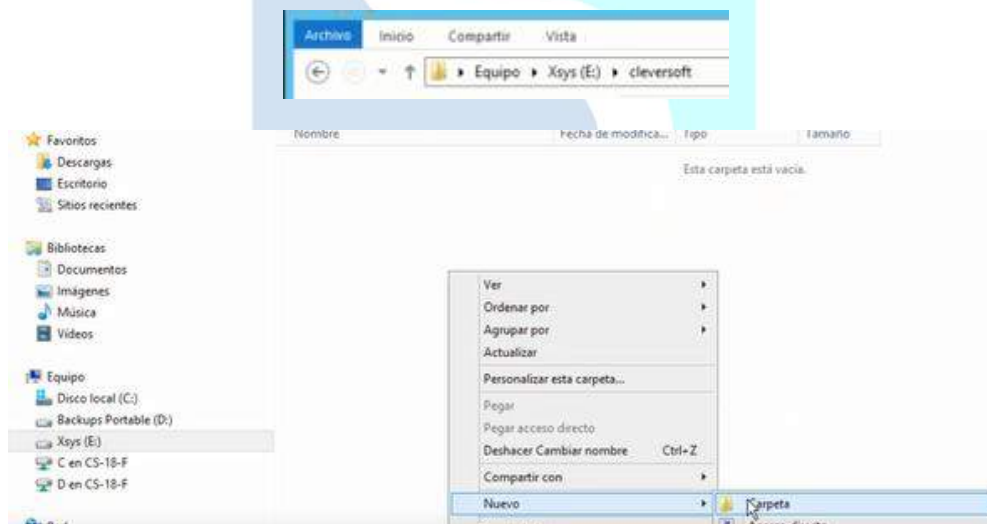
**PASO 1:** Si no existe. Crear en el disco donde se desee realizar la instalación (C:\, E:\, ...), la carpeta cleversoft.

Ejemplo:



**PASO 2:** Si no existe. Crear en la carpeta cleversoft ("unidad elegida":\ cleversoft \), la carpeta sys.

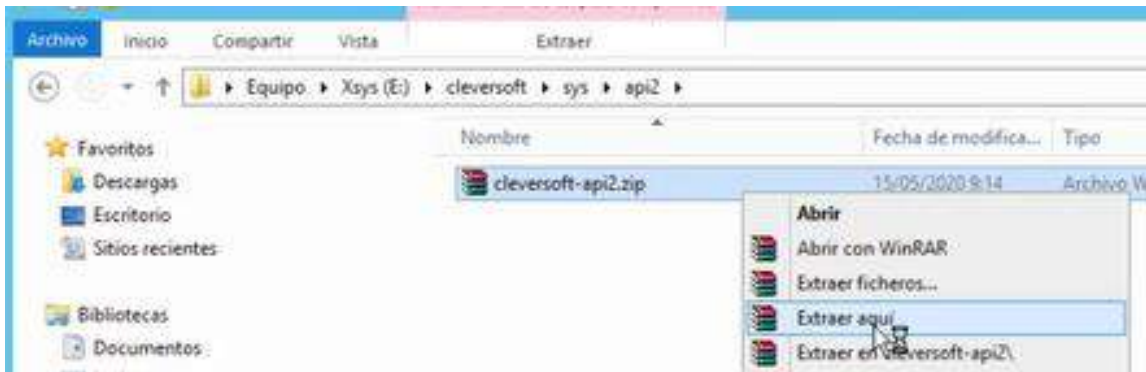
Ejemplo:



**PASO 3:** Descargar el archivo "cleversoft-api2.zip" de la dirección <https://cleversoftsolutions.com/kiosk/downloads/cleversoft-api2.zip>

**PASO 4:** Si no existe. Crear en la carpeta sys ("unidad elegida":\ CleverSoft \sys\), la carpeta api2

**PASO 5:** Descomprimos el archivo descargado dentro de la carpeta API2 ("unidad elegida":\ CleverSoft \sys\api2)



**PASO 6:** Ir a la página de licencias (<https://cleversoftsolutions.com/lic/login.php>), Username: Consulta

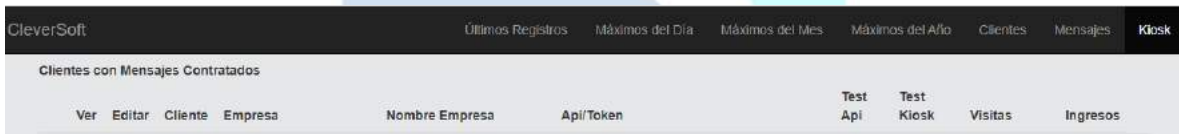
### Control de Licencias CleverSoft Login

**Username:**

**Password:**

Recordar Ingreso

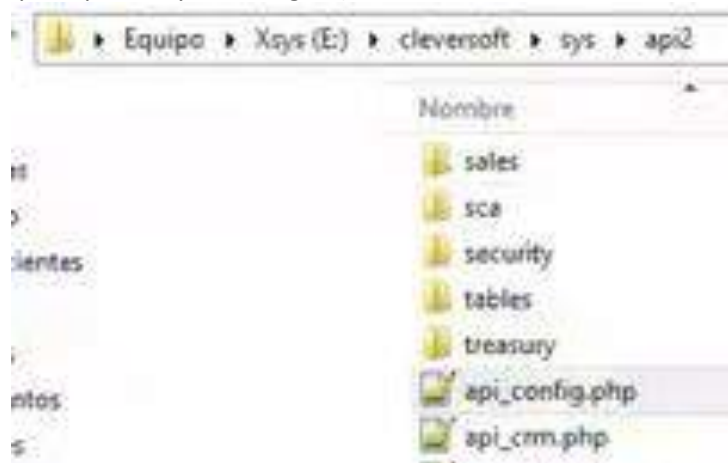
Ir a la pestaña kiosk



Ubicar al cliente y copiar el token

**Nota:** si el cliente no tiene token, crearlo como se indica en el apartado activación licencias para clientes

**PASO 7:** Copiamos, dentro de la carpeta API2 (“unidad elegida”:\ CleverSoft \sys\api2), el archivo api\_config.php (Este archivo se copia de la api de prueba y se configura)



**PASO 8:** Configuramos el archivo `api_config.php`, los parámetros

`$nombre_cliente`

`$db_master`

`$id_cliente`

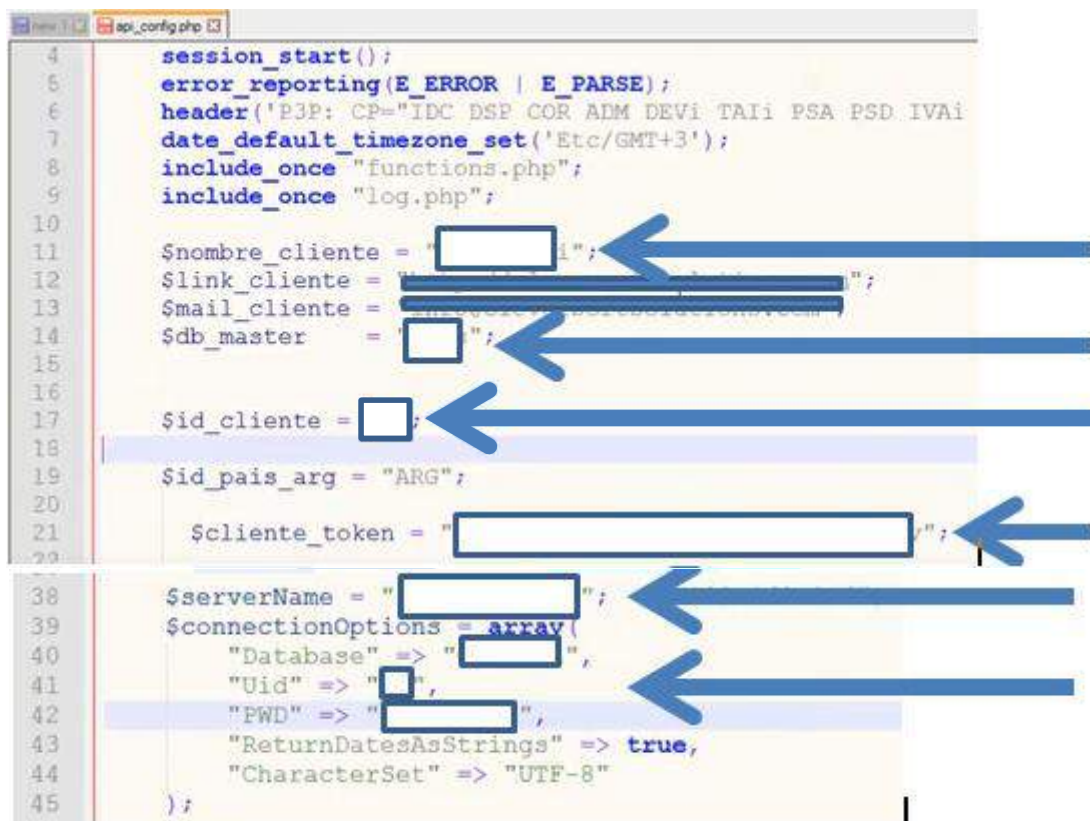
`$cliente_token`

`$ServerName`

Database

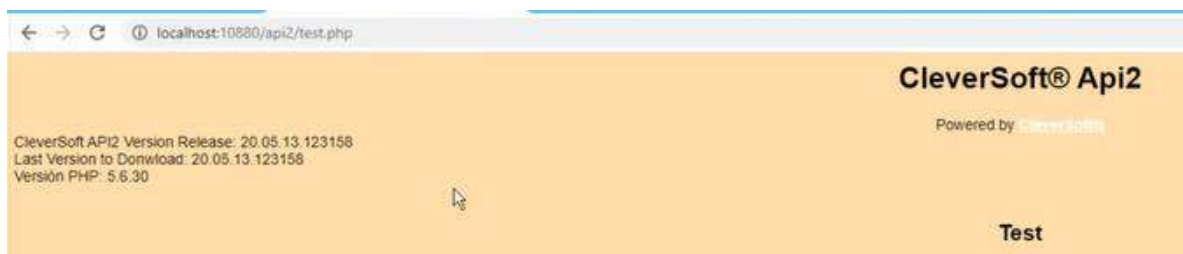
Uid

PWD



```
4 session_start();
5 error_reporting(E_ERROR | E_PARSE);
6 header('P3P: CP="IDC DSP COR ADM DEVI TAIi PSA PSD IVAi
7 date_default_timezone_set('Etc/GMT+3');
8 include_once "functions.php";
9 include_once "log.php";
10
11 $nombre_cliente = " ";
12 $link_cliente = " ";
13 $mail_cliente = " ";
14 $db_master = " ";
15
16
17 $id_cliente = " ";
18
19 $id_pais_arg = "ARG";
20
21 $cliente_token = " ";
22
23
24
25
26
27
28 $serverName = " ";
29 $connectionOptions = array(
30     "Database" => " ",
31     "Uid" => " ",
32     "PWD" => " ",
33     "ReturnDatesAsStrings" => true,
34     "CharacterSet" => "UTF-8"
35 );
```

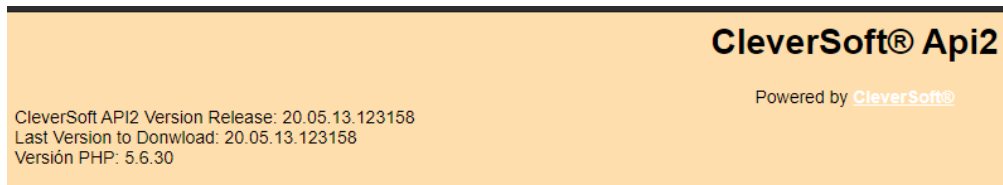
**PASO 9:** Comprobamos que la api quedase bien instalada, para ello desde el navegador en la pc donde se está instalando se coloca `localhost:10880/api2/test.php`



Se debe visualizar la página para prueba de las funciones del api

## Uso de la página Test de la API

Para realizar la verificación de la API, se puede hacer uso de la página de prueba, para ello, ingresamos a <http://Nombre o dirección del servidor/api2/test.php> si estamos desde el server del cliente donde se instaló la API <http://localhost:10880/api2/test.php>



Desde esta página pueden ir evaluando aspectos del api



En cada uno de los módulos solo necesitarás la información precisa la cual se colocará en data y el token generado para el cliente. Por ejemplo:

Para verificar si está funcionando la API vamos a su caja de prueba, solo modificamos en data el número del cliente y colocamos el token de él.

api_test:
<input type="text" value="1"/>
data:
<input type="text" value='{"id_cliente":"1"}'/>
token:
<input type="text" value="nAJEz4kBR0WXBWArNOKjkEjkAmVf5s2"/>
debug:
<input type="text" value="1"/>
<input type="button" value="Enviar"/>

Presionamos enviar, y obtendremos el resultado

```
{"data":{"id":100},"status":200,"message":"Prueba correcta"}
```

De no estar funcionando o haber introducido algo mal, el resultado será por ejemplo para token erróneo

```
{"data":"","status":400,"message":"Token Invalido"}
```



## Configuración de las Dirección WEB para el cliente

Para configurar las direcciones web, se deben seguir los siguientes pasos:

**PASO 1:** Configurar dentro de proveedor de hosting un subdominio al cliente dentro del dominio “miclub.info” directorio raíz /public\_html/kiosk (Para uso del Kiosk)



The screenshot shows a web form titled "Crear un subdominio". It has three main sections: "Subdominio" with a text input field containing "Nombre del cliente"; "Dominio" with a dropdown menu showing "miclub.info"; and "Directorio raíz" with a text input field containing "/public\_html/kiosk". At the bottom, there is an orange "Crear" button.

**PASO 2:** Configurar dentro de proveedor de hosting un subdominio al cliente dentro del dominio “cleversoftsolutions.com” directorio raíz /public\_html/sys (Para uso del ERP WEB y Consultas)



The screenshot shows a web form titled "Crear un subdominio". It has three main sections: "Subdominio" with a text input field containing "Nombre del cliente"; "Dominio" with a dropdown menu showing "cleversoftsolutions.com"; and "Directorio raíz" with a text input field containing "/public\_html/sys". At the bottom, there is an orange "Crear" button.

**PASO 3:** Ir a la carpeta Kiosk y ubicar el archivo “config\_hosts.php”

- ▼ kiosk
  - > .well-known
  - > backUp
  - > cache
  - > cgi-bin
  - > css
  - > downloads
  - > fonts
  - > gire
  - > img
  - > js
  - > lib
  - > log
  - > modules
  - > nuevosocio
  - > operaciones
  - > utils
  - .htaccess
  - abonos.php
  - api\_test.php
  - clave.php
  - cleversoft-kiosk.zip
  - config.php
  - config\_hosts.php

**PASO 4:** Ubicar en el archivo la sesión de case y agregar el que corresponde

```
switch ($host) {  
    case 'nombredelaempresa.miclub.info':  
        $id_empresa = 1;  
        $empresa = "prueba";  
        break;  
    case 'otraempresa.miclub.info':  
        $id_empresa = 2;  
        $empresa = "otra";  
        break;  
}
```

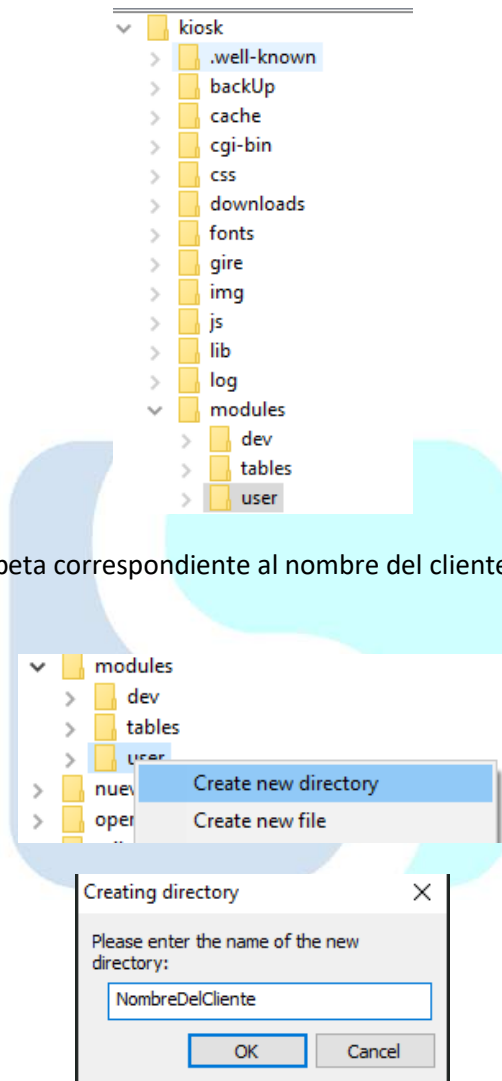


## Alta de user del cliente

Para dar de alta una user particular al cliente para poder configurar los aspectos exclusivos de este debemos seguir los siguientes pasos:

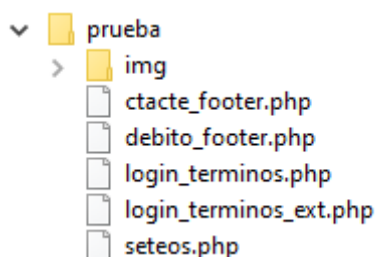
**PASO 1:** Ingresar al servidor de cleversoft-101 e ir a la ruta del servidor apache.

**PASO 2:** Ubicar la carpeta Kiosk y dentro de esta ir a modules/user

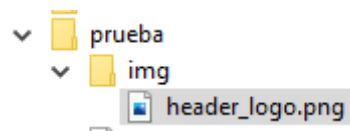


**PASO 3:** Dentro de user, crear la carpeta correspondiente al nombre del cliente (Mismo nombre que usa en la configuración del ERP)

**PASO 4:** Copiar el contenido de la carpeta prueba (ubicada en modules/user) a la carpeta del cliente creada en el paso anterior



**PASO 5:** En la ruta modules/user/"nombredelcliente"/img, subir el logo del cliente con el nombre "header\_logo.png"



**Nota:** La imagen por preferencia debe ir sin fondo y un tamaño de 180 x 180 pixeles.

**PASO 6:** En el archivo seteos.php ubicar el menú principal y configurar en 1 las funciones a activar

```
/****** Menú Principal

$flag_index_ctacte = 1;
$flag_index_actividades = 1;
$flag_index_debito = 1;
$flag_index_crm = 1;
$flag_index_turnos = 1;
$flag_index_datos = 1;
```

**Nota:** De ser necesario configurar si el resumen de la cuenta debe estar por separado por unidad de negocio con las siguientes líneas de código

```
/* //Separación de Resumen de Cta. por unidad de negocios

$flag_index_deuda_separada_items[] = array( "id_unid_negocio"=>"CLUB",
                                             "label"=>"Club",
                                             "id_tipo_cbte_rec"=>"COB"
                                             );

$flag_index_deuda_separada_items[] = array( "id_unid_negocio"=>"INST",
                                             "label"=>"Instituto",
                                             "id_tipo_cbte_rec"=>"RECI"
                                             );

*/
```

## Configurar método de pago

Para configurar el método de pago que utilice el cliente, se deben seguir los siguientes pasos:

**PASO 1:** Abrir el archivo seteos.php del cliente ubicado en la ruta modules/user/"nombredelcliente"/

**PASO 2:** Dentro del archivo ubicar o crear la sección de pagos

```
/****** Sección Pagos *****/  
$flag_medio_pago = ""; // NO SPS GIRE NPS MP
```

**PASO 3:** Configurar según método de pago con la siguiente estructura rellenando las variables entre las "", con la información suministrada por el cliente y según el método a utilizar

### GIRE

```
$flag_medio_pago = "GIRE"; // NO SPS GIRE NPS MP  
  
$GIRE_PRIVATE_KEY           = "";  
$GIRE_GS_USERNAME           = "";  
$GIRE_GS_PASSWORD           = "";  
$GIRE_GS_MERCHANT_ID       = "";  
$GIRE_GS_AMBIENTE           = "";  
$GIRE_GS_URL_HOME           = "";  
$GIRE_GS_URL_CONFIRM        = "";  
  
// Private  
$decidir_key                 = "";  
// Public  
$decidir_client_key         = "";
```

### MP

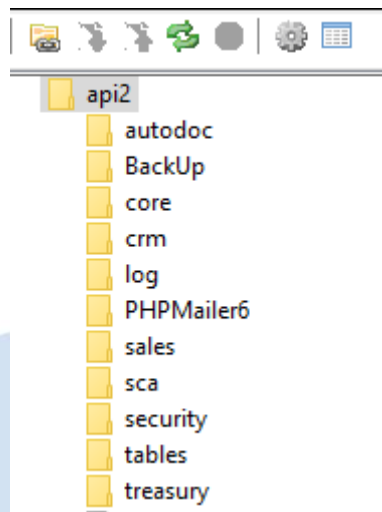
```
$flag_medio_pago = "MP"; // NO SPS GIRE NPS MP  
  
$MP_PUBLIC_KEY              = "";  
$MP_ACCESS_TOKEN            = "";  
$MP_CLIENT_ID               = "";  
$MP_CLIENT_SECRET           = "";
```

## Creación de una función

Para crear una función o método en la API, tomando en consideración que todos se manejan en archivos aparte, deben seguirse los siguientes pasos:

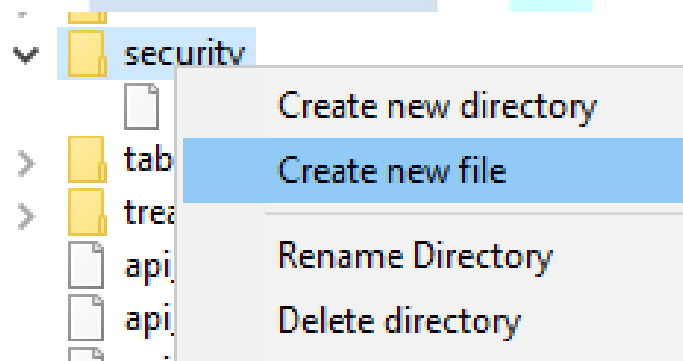
**PASO 1:** Verificamos a que modulo va a pertenecer principalmente la nueva función o método.

**PASO 2:** Ubicar dentro de la carpeta contenedora de la API, la carpeta del módulo al que va a pertenecer nuestra función o método.



**Nota:** De no existir la carpeta para el modulo, crear la misma en el contenedor de la API

**PASO 3:** Crear archivo .php dentro de la carpeta del módulo (nombres siempre en inglés)



Creating file



Please enter the name of the new file:

name.php

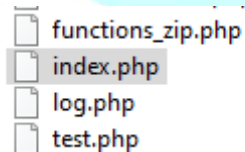
OK

Cancel

**PASO 4:** El archivo debe comenzar con la siguiente estructura para y a partir de ella ir armando el query.

```
1 <?
2
3 # =====
4 #
5 #
6 # Nombre de la función
7 #
8 # =====
9 #
10 # data (POST)
11 # {"Parametro":"1"}
12 #
13 # =====
14
15
16 $data = json_decode($_POST["data"], true);
17 $jsonerror = json_last_error();
18
19 $data_Parametro = trim($data["Parametro"]);
20
21 // verifica si el json es valido
22 if ($jsonerror){
23     $res = array("data" => "", "status" => 500, "message" => "Invalid JSON");
24     echo json_encode($res);
25     exit;
26 }
27
```

**PASO 5:** Ubicar en la carpeta contenedora de la API el archivo index.php



- functions\_zip.php
- index.php
- log.php
- test.php

**PASO 6:** Dentro del archivo index.php ubicar donde están configuradas las llamadas de un módulo y agregar la nueva línea

```
/*Example Module*/
if ($_POST["example"]==1) { include_once __DIR__ ."/carpeta modulo/name.php"; }
```

**PASO 7:** Para chequear la función en el archivo test.php ubicado en la carpeta contenedora de la API, agregar copiando de una allí existente un cuadro de validación dentro del nivel del módulo.

## Como insertar el AutoDoc (Documentación)

**PASO 1:** Para insertar el autodoc lo único que debe realizarse, es incluir al principio de cada archivo la siguiente estructura, incluyendo las líneas de inicio y fin de comentario.

```
/* clever-autodoc
<clever-autodoc>

<file name="Ruta donde esta el archivo">
<name> Nombre del archivo con su extensión ej. nombre.php </name>
<last_update>Fecha de actualización en el siguiente formato 12/06/2019 17:10Hs</last_update>

<updates>
  <item>
    <date>Fecha de actualización en el siguiente formato 12/06/2019 17:10Hs</date>
    <description>Actividad realizada</description>
  </item>
</updates>

<functions>
  <item>
    <name>Nombre de la función</name>
    <description>Descripción de la función</description>
  </item>

  <parameters>
    <item>
      <name>Nombre del parametro</name>
      <type>Tipo del parametro ej. VARCHAR(5000)</type>
      <description>Descripción del parametro</description>
    </item>
  </parameters>

  <return>
    <status_posibles>
      <item>
        <id>200</id>
        <description>OK</description>
      </item>
      <item>
        <id>400</id>
        <description>Token Invalido</description>
      </item>
      <item>
        <id>401</id>
```



```
<description>JSON Data Invalido</description>
</item>
<item>
<id>402</id>
<description>Valores Parametros Data Invalido</description>
</item>
<item>
<id>403</id>
<description>Error Interno SQL Server</description>
</item>
<item>
<id>405</id>
<description>Error sin descripcion</description>
</item>
</status_posibles>
<data>
<item>
<name>Nombre de la variable</name>
<type>Tipo de la variable Ej. VARCHAR(14)</type>
<description>Descripción de la variable</description>
</item>
</data>
</return>
</functions>

</file>

</clever-autodoc>
*/
```

**NOTA:** Dentro de cada etiqueta debe colocarse la información correspondiente, en el caso de lo que va dentro de las etiquetas **<updates>** **</updates>** Debe colocarse un item por cada actualización y siempre deben estar de forma descendente siendo la primera en lectura la fecha mas actual.

Con respecto a la etiqueta **<data>** **</data>** Debe colocarse un item por cada variable que este incluida dentro de la función.