

# Estructura de Código en API

las paginas de la API se desarrollaron en PHP con una conexion PDO a la BD, en el codigo fuente de las pagina vamos a encontrar en su mayoria la misma estructura que enseñaremos a continuacion, continua leyendo para aprender a crear nuevas acciones.

## Acciones de Pagina Ejemplo

- Eliminar
- Crear
- Ver
- Modificar
- Categorías
- Historial

```
# -----  
----- #  
# Author: FelipeGomez  
# Author URL: http://demedallo.com & http://feliphegomez.inf  
o  
# License: Creative Commons Attribution 3.0 Unported  
# License URL: http://creativecommons.org/licenses/by/3.0/  
# -----  
----- #  
require_once("autoload.php");  
  
$jsonFinal = new stdClass();
```

```
$jsonFinal->error = true;
$jsonFinal->message = "Parametros invalidos consulte la docum
entacion de la API.";

if(isset($_POST) && count($_POST)>0){ $data = ($_POST); }
elseif(isset($_GET) && count($_GET)>0){ $data = ($_GET); }
elseif(isset($_DELETE) && count($_DELETE)>0){ $data = $_DELET
E; }
else{ $data = array(); };

if(isset($data['accesstoken']))){
    $checkToken = chehear_AccessToken_CCyUser($data['accesst
oken']);
    if($checkToken==false){
        $jsonFinal->error_message = "Accesstoken Invalido";
    }else{
        if(isset($data['action']) && $data['action'] == 'dele
te'){
            //Codigo para eliminar su contenido
        }else if(isset($data['action']) && $data['action'] ==
'create'){
            //Codigo para crear su contenido
        }else if(isset($data['action']) && $data['action'] ==
'view'){
            //Codigo para ver su contenido
        }else if(isset($data['action']) && $data['action'] ==
'change'){
            //Codigo para modificar su contenido
```

```

        }elseif(isset($data['action']) && $data['action'] ==
"categories"){
            // Codigo para ver, crear, modificar y eliminar l
as categorias de su contenido
        }elseif(isset($data['action']) && $data['action'] ==
"history"){
            // Codigo para ver el historial de su contenido
        }
    }
}
}else{
    $jsonFinal->message = "Falta el accesstoken";
};

#FINAL
header('Content-Type: application/json');
echo json_encode($jsonFinal,JSON_PRETTY_PRINT);
return json_encode($jsonFinal,JSON_PRETTY_PRINT);

```

## Lo Basico

- `chehear_AccessToken_CCyUser( {accessToken-user} )`
- `$data[ {campo-key} ]`
- `$checkToken[ {campo-key-user-check} ]`

### `chehear_AccessToken_CCyUser()`

Se utiliza para comprobar el token de acceso a Forma2 para el usuario que realiza la consulta o peticion al servidor, si deseas mas informacion

consulta el manual tecnico.

Esta conformada por una cadena codificada en Base64 con los datos:

- cedula
- login / user / nick / username
- fecha de creacion del accesstoken

**`$data[ {campo-key} ]`**

Los campos o fields recibidos por GET / POST son almacenados en la variable `$data` para acceder a uno se debe utilizar el metodos Array, ejemplo: `$data[ 'check' ]` si se desea utilizar el campo de check de la consulta.

GET /forma2/api/v1.0/login.php?

`check=YWZnb21lbWF8MTAzNTQyOTM2MHwyMDE4LTAyLTI4IDEwOjU10jQy`

**`$checkToken[ {campo-key-user-check} ]`**

Despues de que el sistema compruebe los datos del accesstoken so almacenados en la variable `$checkToken` , para acceder a sus componentes lo manajeramos como un array y en caso de acceder a un elemento anidado se maneja como una stdClass, te mostraremos ahora un ejemplo de array.

- Para acceder al nombre de la persona que envia el token de acceso se haria de la siguiente manera -> `$checkToken[ 'nombre' ]`

Para conocer todos los datos a los que podemos acceder puedes consultar la guia de Tokens de Acceso.

## Comprobar Permisos

Para comprobar los permisos del token de acceso recibido se deben hacer desde la variable `$checkToken['permisos_cargos']` y en caso de ser anidados te mostraremos un ejemplo complejo para que no tengas inconvenientes, en el ejemplo se va a comprobar los permisos para acceder al historial del quiz.

```
if(isset($checkToken['permisos_cargos']->quiz->history) && $checkToken['permisos_cargos']->quiz->history == true){  
    $jsonFinal->message = "Tienes Acceso al historial";  
}else{  
    $jsonFinal->message = "No tienes acceso al historial";  
};
```

## Nueva consulta de datos

Dentro de la API podemos utilizar la funcion `datosSQL( {consulta-sql} )` para acceder de forma rapida a una cosulta tipo **SELECT**:

- En el ejemplo se va utilizar *"SELECT id, firstname, lastname FROM MyGuests"*.

## Nueva creacion o modificacion de datos

Dentro de la API podemos utilizar la funcion `crearSQL( {consulta-sql} , { datos-array } )` para insertar o actualizar informacion de manera rapida, las consultas son **INSERT & UDPATE**:

- En el ejemplo se va utilizar *“INSERT INTO MyGuests (firstname, lastname, email) VALUES (?, ?, ?)”*.

```
$change = crearSQL("INSERT INTO MyGuests (firstname, lastname
, email)
VALUES (?, ?, ?)", array('John', 'Doe', 'john@example.com'))
;
if($change->error == false){
    $jsonFinal->error = false;
    $jsonFinal->last_id = $change->last_id;
    $jsonFinal->message = "Creado con exito!.";
}else{
    $jsonFinal->error_message = "Error creando.";
};
```

- En el ejemplo se va utilizar *“UPDATE MyGuests SET lastname=? WHERE id=2”*.

```
$change = crearSQL("UPDATE MyGuests SET lastname=? WHERE id=2
", array("Doe"));
if($change->error == false){
    $jsonFinal->error = false;
    $jsonFinal->message = "lastname actualizado con exito!";
}else{
    $jsonFinal->error_message = "Error actualizando el lastna
me";
};
```

# Respuestas API

## OK - Respuesta satisfactoria

Si la consulta fue exitosa se vera en la mayoría de las veces una respuesta similar a la que te enseñaremos a continuacion.

```
{
  "error": false,
  "message": "{ Mensaje de consulta cargada con exito. }",
  "fields": {
    "action": "{ view / create / delete / history / change }"
    "{ field-1-key }": "{ field-1-value }"
  },
  "data": "{informacion de la peticion}"
}
```

## Error - Campos / Fields Invalidos

En el cuyo caso que los campos que enviemos no sean reconocidos o no lleguen de manera correcta, la respuesta es:

```
{
  "error": true,
  "message": "Parametros invalidos consulte la documentacion de la API."
}
```

## Error - Token de Acceso Invalido

En caso de que el accesstoken sea invalido dara como resultado la siguiente respuesta en codigo JSON.

```
{  
  "error": true,  
  "message": "Accesstoken Invalido."  
}
```