



EBook Gratis

APRENDIZAJE sugarcrm

Free unaffiliated eBook created from
Stack Overflow contributors.

#sugarcrm

Tabla de contenido

Acerca de	1
Capítulo 1: Empezando con Sugarcrm	2
Observaciones.....	2
Examples.....	2
Instalación o configuración.....	2
Capítulo 2: API REST v10	3
Introducción.....	3
Examples.....	3
Iniciar sesión.....	3
Operaciones básicas de CRUD.....	4
Creación de puntos finales personalizados.....	5
Creditos	8

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [sugarcrm](#)

It is an unofficial and free sugarcrm ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official sugarcrm.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con Sugarcrm

Observaciones

Esta sección proporciona una descripción general de qué es sugarcrm y por qué un desarrollador puede querer usarlo.

También debe mencionar cualquier tema importante dentro de Sugarcrm, y vincular a los temas relacionados. Dado que la Documentación para sugarcrm es nueva, es posible que deba crear versiones iniciales de los temas relacionados.

Examples

Instalación o configuración

Instrucciones detalladas sobre cómo configurar o instalar Sugarcrm.

Lea Empezando con Sugarcrm en línea:

<https://riptutorial.com/es/sugarcrm/topic/10038/empezando-con-sugarcrm>

Capítulo 2: API REST v10

Introducción

La API RESTful de SugarCRM se actualizó a la versión 10 en las versiones de SugarCRM 7.x.

Como las versiones 7.x se crean como una aplicación de una sola página, la API es el principal medio de comunicación con el servidor para recuperar datos. Las integraciones de terceros también pueden vincularse y acceder a los datos. La API también se puede ampliar para proporcionar puntos finales personalizados, implementando la funcionalidad deseada.

[Documentos oficiales](#)

Examples

Iniciar sesión

Puede usar este ejemplo como una llamada desde el código, oa través de un cliente REST como [Postman](#) :

POST <https://YOURSITE.com/rest/v10/oauth2/token>

```
{
  "grant_type": "password",
  "client_id": "sugar",
  "client_secret": "",
  "username": "your_username",
  "password": "your_P@$sw0rd",
  "platform": "api"
}
```

Para profundizar en los detalles anteriores, representan:

"grant_type": "password" : este es el método de inicio de sesión. En este caso, estamos usando una combinación de nombre de usuario y contraseña, por lo que pasamos a través de "password"

"client_id": "sugar" : esto indica que nos estamos autenticando a través del cliente "sugar" estándar de nombre / contraseña. Otros clientes como "support_portal". Esto también puede hacer referencia a una clave OAuth personalizada dentro de la aplicación.

"client_secret": "" : está en blanco, ya que estamos usando el ID de cliente de "sugar", pero si está usando un cliente personalizado, este es el secreto asociado con el cliente.

"username": "your_username" - Tu nombre de usuario

"password": "your_P @ \$ sw0rd" - Tu contraseña

"plataforma": "api" : las plataformas típicas utilizadas en Sugar son "base", "mobile" y "portal". Por razones de seguridad, si el mismo usuario inicia sesión en la misma plataforma simultáneamente, el usuario saldrá de la sesión anterior. He usado "api", pero esto podría ser "myawesomesugarintegration" si quisieras.

Tras una solicitud exitosa, el servidor devuelve un objeto que incluye un "access_token", es decir:

```
"access_token": " abcdef01-2345-6789-0abc-def012345678 "
```

Para cada solicitud posterior a la API, debe incluir este token para autenticarse.

Esto se puede lograr agregando el encabezado "OAuth-Token" con el valor access_token a cualquier otra solicitud.

Operaciones básicas de CRUD

Primero, asegúrese de [iniciar sesión](#) y agregue access_token al encabezado de su solicitud.

En este ejemplo, vamos a hacer algunas operaciones básicas para acceder a los registros. Estos utilizan el módulo de Cuentas como ejemplo, pero otros módulos estándar y personalizados (ej. Leads, Contactos, Oportunidades) se comportan de la misma manera.

Crear

Para crear una cuenta, debemos realizar una solicitud de publicación al punto final de Cuentas, con los detalles que deseamos agregar. En caso de éxito, devuelve un objeto que contiene el ID del nuevo registro y los datos actuales.

```
POST https://YOURSITE.com/rest/v10/Accounts
{
  "name": "My New Account"
}
```

Leer

Primero, recuperaremos el registro que acabamos de crear, para mí el ID fue "9174c58c-409c-11e7-bfdf-00163ef1f82f" para recuperar toda la información del registro, hacemos lo siguiente:

```
GET https://YOURSITE.com/rest/v10/Accounts/9174c58c-409c-11e7-bfdf-00163ef1f82f
```

¡Seguro que es un objeto grande! ¿Qué hay de ver el nombre y la fecha en que lo creé?

```
GET https://YOURSITE.com/rest/v10/Accounts/9174c58c-409c-11e7-bfdf-00163ef1f82f?fields=name,date_entered
```

Mucho mejor. Pero, ¿qué sucede si ya tengo miles de cuentas en el sistema y no he logrado recordar el GUID?

```
GET https://YOURSITE.com/rest/v10/Accounts?fields=name,date_entered&filter[0][name]=My New Account
```

Actualizar

Entonces, ¿y si decidiera que quería cambiar algo en la aplicación? ¿Qué hay de cambiar el nombre y agregar una descripción?

```
PUT https://YOURSITE.com/rest/v10/Accounts/9174c58c-409c-11e7-bfdf-00163ef1f82f
{
  "name": "My Updated Account",
  "description": "Updated via REST API"
}
```

Borrar

De acuerdo, todo está bien, pero aclaremos esto antes de poner datos reales en:

```
DELETE https://YOURSITE.com/rest/v10/Accounts/9174c58c-409c-11e7-bfdf-00163ef1f82f
```

Creación de puntos finales personalizados

Una de las características de SugarCRM 7.x es poder agregar y extender fácilmente puntos finales personalizados para lograr lo que necesita.

En este ejemplo, crearemos un par de puntos finales personalizados para devolver algunos datos sobre la solicitud.

Este archivo personalizado se está colocando en `custom/clients/base/api/DescriptionAPI.php`.

```
<?php

if(!defined('sugarEntry') || !sugarEntry) die('Not A Valid Entry Point');

//You need to inherit the SugarApi Class to
class DescriptionApi extends SugarApi
{
    public function registerApiRest()
    {
        return array(

            //Define a key for the array
            'DescribeRequest' => array(

                //Array of the acceptable kinds of requests for this method
                'reqType' => array('GET', 'POST', 'PUT', 'DELETE'),

                //If true, anyone can access. If false, only authenticated users.
                'noLoginRequired' => true,

                //Here is the path to access the endpoint, in this case: Describe/Request
                'path' => array('Describe', 'Request'),

                //Specify an empty string for the path variables
```

```

        'pathVars' => array('', ''),

        //method to call
        'method' => 'DescribeMyRequest',

        //A small description, displayed in rest/v10/help page
        'shortHelp' => 'Describes your Request Method',

        //Further help, displayed when drilling down into the help page
        'longHelp' => 'custom/clients/base/api/help/DescribeRequestHelp.html',
    ),
    //Here's another entry with some more in depth information
    'DescribeIncludingArgs' => array(
        'reqType' => array('GET', 'POST', 'PUT', 'DELETE'),
        'noLoginRequired' => true,

        //This time, we'll include a third element with a ?
        //So now the path is Describe/Request/{dataFromURL}
        'path' => array('Describe', 'Request', '?'),

        //Here, we specify the key for accessing that data within the function
        'pathVars' => array('', '', 'dataFromURL'),

        'method' => 'DescribeMyRequestIncludingArguments',
        'shortHelp' => 'Describes the request you sent, including method, URL
parameters, and request body',
        'longHelp' => 'custom/clients/base/api/help/DescribeIncludingArgsHelp.html',
    ),
);

}

/**
 * Your custom logic goes in here.
 */
public function DescribeMyRequest($api, $args)
{
    //Find out the request method sent
    $requestType = $_SERVER['REQUEST_METHOD'];

    return "You sent a $requestType request.";
}

/**
 * Here is the second function
 */
public function DescribeMyRequestIncludingArguments($api, $args)
{
    //Find out the request method sent
    $requestType = $_SERVER['REQUEST_METHOD'];

    //Get the data included in the URL parameter
    $data = $args['dataFromURL'];

    //Read from the request body
    $body = file_get_contents('php://input');

    return "You sent a $requestType request including the header argument: ` $data ` and the
body: ` $body `";
}

```



```
}
```

Después de agregar este archivo, deberá realizar una Reparación y Reconstrucción para que Sugar registre su punto final correctamente.

Después, si navega a `rest / v10 / Describe / Request` en su navegador, debería ver:

"Enviaste una solicitud GET".

Ahora, si usa un cliente REST para enviar una solicitud POST, con algunos datos, por ejemplo:

`POST rest/v10/Describe/Request/Stuff` con el cuerpo `{"key": "value"}` debe recibir:

"Envío una solicitud POST que incluye el argumento del encabezado: `Stuff` y el cuerpo: `{\"key\": \"value\"}` "

Lea API REST v10 en línea: <https://riptutorial.com/es/sugarcrm/topic/10043/api-rest-v10>

Creditos

S. No	Capítulos	Contributors
1	Empezando con Sugarcrm	Community
2	API REST v10	Reisclef