



# SugarCRM REST Bundle

---

Developer documentation

---

Author: **NeoArno.com** ([me@neoarno.com](mailto:me@neoarno.com))

Website: **blog.neoarno.com**

Version: **0.2**

Date: **August 2013**

## Pre-requisite

---

To be done...

## Installation

---

Add SugarRESTRBundle in your composer.json:

```
{
    "require": {
        "neo-arno/sugar-rest-bundle": "dev-master",
        ...
    }
}
```

Update your application

```
$ php composer.phar update neo-arno/sugar-rest-bundle
```

Now the bundle is installed in your *vendor/neo-arno/sugar-rest-bundle* directory

Enable the bundle in the *AppKernel.php* file :

```
$bundles = array(
    // ...
    new NeoArno\sugarRESTRBundle\sugarRESTRBundle(),
);
```

This bundle use a REST client bundle by default : *"leaseweb/api-caller-bundle"*

**But you can use the REST API bundle you want ;-) see bellow for more information**

Now just initialize the REST URL for your SugarCRM instance in the file : *Resources/config/services.yml*

```
// ...
rest_url: "http://<your-sugarcrm-instance-url>/service/v4_1/rest.php"
api_version: "4.1"
```

rest\_url: REST URL to your instance. Don't forget to change your *.htaccess* file if necessary.

api\_version : parameter is reserved for a futur use

## Objects

---

### Server

To use this object just create an instance , for example in your controller :

```
//...  
$sugarserver = $this->get('sugarserver');  
//...
```

The Server object is dedicated to specific server functions to retrieve configuration, parameters, ...

Name	<b>GetServerInfo</b>
Description	<p>Gets server info. This will return information like version, flavor and gmt_time.</p> <p>This function does not need a valid session ID.</p>
Parameters	-
Return	<p><i>SugarRESTResult</i> object with a ResultArray :</p> <ul style="list-style-type: none"><li>▪ <i>flavor</i> : SugarCRM version type (PRO,ENT,CE,...)</li><li>▪ <i>version</i> : version number of the instance</li><li>▪ <i>gmt_time</i> : the current GMT Time of the server (in the format 'Y-m-d H:i:s')</li></ul>
Example	<pre>\$sessionID = ""; \$resultObj = \$this-&gt;get('sugarserver')                 -&gt;GetServerInfo(\$sessionID);</pre>

Name	<b>GetModuleLayout</b>
Description	Retrieve the layout metadata for a given module given a specific type and view
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : The name of the module(s) to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$view</i> : the view(s) requested. Current supported types are "edit", "detail", "list", and "subpanel". By default : array('edit')</p> <p><i>\$type</i> : the type(s) of views requested. Current supported types are 'default' (for application) and 'wireless' (for mobile). By default : array('default')</p> <p><i>\$acl_check</i> : verify the ACL for module and requested view for the session user. By default : "true"</p> <p><i>\$md5</i> : if true, return only the MD5 number for requested module and view. By default, the value is <i>false</i>.</p>
Return	<p><i>SugarRESTResult</i> object with a <i>ResultArray</i> :</p> <ul style="list-style-type: none"> <li>for each module, each type and each view requested return the metadata array (look at <i>SugarCRM</i> document or in the metadata directory of the module)</li> </ul>
Example	<pre> \$resultObj = \$this-&gt;get('sugarserver')     -&gt; GetModuleLayout(\$sessionID,                         array("Accounts"),                         array("detail"),                         array("default"),                         true,                         false); </pre>

Name	<b>GetModuleFields</b>
Description	Retrieve vardef information on the fields of the specified bean.
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : The name of the module to return record from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$fields</i> : Optional, if passed then retrieve vardef information on these fields only.</p>
Return	<p><i>SugarRESTResult</i> object with a ResultArray :</p> <ul style="list-style-type: none"> <li>▪ <i>module_fields</i> : The array vardef information on the selected fields.</li> <li>▪ <i>link_fields</i> : The array vardef information on the link fields</li> </ul>
Example	<pre>\$resultObj = \$this-&gt;get('sugarserver') -&gt;GetModuleFields(\$sessionID,"Accounts");</pre>

Name	<b>GetLanguageDef</b>
Description	Retrieve the module(s) language array based on the current session user default language.
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : An array of module(s) name(s). This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$md5</i> : if true, return only the MD5 number for requested module and view. By default, the value is <i>false</i></p>
Return	<p><i>SugarRESTResult</i> object with a ResultArray :</p> <ul style="list-style-type: none"> <li>▪ <i>for each module, return an array with the language metadata array</i></li> </ul>
Example	<pre>\$resultObj = \$this-&gt;get('sugarserver') -&gt;GetLanguageDef(\$sessionID, array("Accounts","Contacts"));</pre>

## User

To use this object just create an instance , for example in your controller :

```
//...  
$sugaruser = $this->get('sugaruser');  
//...
```

Name	<b>Connect</b>
Description	Logs user into the SugarCRM application
Parameters	<p><i>\$user</i> : user login to the application</p> <p><i>\$password</i> : password associated to the user</p> <p><i>\$language</i> : language code for this user ("en_US" by default)</p> <p><i>\$notifyonsave</i> : a boolean when it's true generate an email notification when a record is saved (false by default)</p>
Return	<p><i>SugarRESTResult</i> object with a <i>ResultArray</i> :</p> <ul style="list-style-type: none"><li>▪ <i>"id"</i> is the sessionID</li><li>▪ <i>a name-value array with information on connected user</i><ul style="list-style-type: none"><li>○ <i>user_id</i> : user identification (ex. "1")</li><li>○ <i>user_name</i> : login of the user</li><li>○ <i>user_language</i> : language selected for this user</li><li>○ <i>user_currency_id</i> : currency id and name selected for this user</li><li>○ <i>user_currency_name</i></li><li>○ <i>user_is_admin</i> : Boolean, true if the user is an administrator</li><li>○ <i>user_default_team_id</i> : default team selected for this user</li><li>○ <i>user_default_dateformat</i> : ex: "Y-m-d"</li><li>○ <i>user_default_timeformat</i> : ex. "H:I"</li><li>○ <i>user_number_seperator</i> : example : "."</li><li>○ <i>user_decimal_seperator</i> : ex. ","</li><li>○ <i>mobile_max_list_entries</i></li><li>○ <i>mobile_max_subpanel_entries</i></li></ul></li></ul>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser') -&gt;Connect('my_login','my_password', "en_US", false) ;</pre>

Name	<b>Logout</b>
Description	Logout the current user
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i>
Return	<i>SugarRESTResult</i> object no specific information returned
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')         -&gt;Logout(\$sessionID) ;</pre>

Name	<b>SeamlessLogin</b>
Description	verify if the sessionID is relative to an active session
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i>
Return	<i>SugarRESTResult</i> object if <i>error_nb</i> is equal to 0 the session is already connected if <i>error_nb</i> is equal to 1 the session is not connected
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')         -&gt; SeamlessLogin(\$sessionID) ;</pre>

Name	<b>GetUserID</b>
Description	Return the user_id of the user that is logged into the current session
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i>
Return	<i>SugarRESTResult</i> object resultObj[0] is the current user ID
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')         -&gt; GetUserID(\$sessionID) ;</pre>



Name	<b>GetAvailableModules</b>
Description	Retrieve the list of available modules on the system available to the currently logged in user
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$filter</i> : possible values are</p> <ul style="list-style-type: none"> <li>⇒ <i>all</i> : Return all modules (default value)</li> <li>⇒ <i>default</i> : Return all visible modules for the application</li> <li>⇒ <i>mobile</i> : Return all visible modules for the mobile view</li> </ul>
Return	<p><i>SugarRESTResult</i> object</p> <p>return information on each module</p> <p>(key, name, favorite enabled, acIs)</p>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')         -&gt; GetAvailableModules (\$sessionID, \$filter);</pre>

Name	<b>GetlastViewed</b>
Description	Retrieve a list of recently viewed records by module
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_array</i> : array of requested module. "Home" for all modules</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>return information on each module</p> <p>(key, name, favorite enabled, acIs)</p>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')         -&gt; GetAvailableModules (\$sessionID, \$filter);</pre>

Name	<b>GetUpcomingActivities</b>
Description	Retrieve a list of upcoming activities including Calls, Meetings, Tasks and Opportunities
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i>
Return	<i>SugarRESTResult</i> object Array List of upcoming activities
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser') -&gt; GetUpcomingActivities(\$sessionID); </pre>

Name	<b>GetUserTeamID</b>
Description	Return the ID of the default team for the user that is logged into the current session
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i>
Return	<i>SugarRESTResult</i> object resultObj[0] is the current user default team ID
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser') -&gt; GetUserTeamID(\$sessionID); </pre>

## Bean

To use this object just create an instance, for example in your controller :

```
//...  
$sugarbean = $this->get('sugarbean');  
//...
```

Name	<b>GetEntry</b>
Description	Get specific entry Data based on an ID
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$obj_id</i> : the SugarBean's ID value.</p> <p><i>\$fields_list</i> : A list of the fields to be included in the results. This optional parameter allows for only needed fields to be retrieved.</p> <p><i>\$link_name_to_fields_array</i> : a list of link_names and for each link_name, what fields value to be returned. For ex.'link_name_to_fields_array' =&gt; array(array('name' =&gt; 'email_addresses', 'value' =&gt; array('id', 'email_address', 'opt_out', 'primary_address')))</p> <p><i>\$trackView</i> : should we track the record accessed (default : FALSE)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>Return an array of each fields, values and relationships list</p>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser') -&gt;GetEntry(\$sessionID, "Accounts", "12a27887-f4a0-7f26-74c8-50a47fc41ec9", array("id","name"));</pre>

Name	<b>SetEntry</b>
Description	Update or create a single SugarBean
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$obj_id</i> : the SugarBean's ID value.</p> <p><i>\$name_value_list</i> : The keys of the array are the SugarBean attributes, the values of the array are the values the attributes should have.</p> <p><i>\$trackView</i> : should we track the record accessed (default : FALSE)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>return ID of object with modify fields</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')  -&gt;SetEntry(\$sessionID, "Accounts", array("id" =&gt; "12a27887-f4a0-7f26-74c8-50a47fc41ec9", "description" =&gt; "Modified by REST API ---")); </pre>

Name	<b>GetEntries</b>
Description	Get specific entry Data based on an array of IDs
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$IDs_array</i> : The SugarBeans' IDs array.</p> <p><i>\$fields_list</i> : A list of the fields to be included in the results. This optional parameter allows for only needed fields to be retrieved.</p> <p><i>\$name_value_list</i> : The keys of the array are the SugarBean attributes, the values of the array are the values the attributes should have.</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>return ID of object with modify fields</p>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')                 -&gt;GetEntries(\$sessionID,"Accounts", array("12a27887-f4a0-7f26-74c8-50a47fc41ec9", "7e66852e-8e67-dee3-553d-50a47f00408f"), array("id","name"));</pre>

Name	<b>GetEntriesCount</b>
Description	retrieve number of records in a given module
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$query</i>: a WHERE clause.</p> <p><i>\$deleted</i>: specify whether or not to include deleted records (default : FALSE).</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>result_count - integer - Total number of records for a given module and query</p>
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')                 -&gt;GetEntriesCount(\$sessionID, "Accounts","");</pre>

Name	<b>SetEntries</b>
Description	Update or create a single SugarBean
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$name_value_list</i> : The keys of the array are the SugarBean attributes, the values of the array are the values the attributes should have.</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>'ids' -- Array of the IDs of the beans that was written to (-1 on error)</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                 -&gt;SetEntries(\$sessionID, "Accounts", array("id" =&gt; "12a27887-f4a0-7f26-74c8-50a47fc41ec9", "description" =&gt; "Modified by REST API ---")); </pre>

Name	<b>GetListEntries</b>
Description	Retrieve a list of Beans answer to a specific request
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i> : the name of the module to return records from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$query</i> : SQL where clause without the word 'where'</p> <p><i>\$order_by</i> : SQL order by clause without the phrase 'order by'</p> <p><i>\$offset</i> : The record offset to start from</p> <p><i>\$fields_list</i> : A list of the fields to be included in the results. This optional parameter allows for only needed fields to be retrieved.</p> <p><i>\$name_value_list</i> : The keys of the array are the SugarBean attributes, the values of the array are the values the attributes should have.</p> <p><i>\$max_results</i> : The maximum number of records to return. (default: 10)</p> <p><i>\$deleted</i> : false if deleted records should not be include, true if deleted records should be included (default: FALSE)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>'result_count' : The number of records returned</p> <p>'next_offset' : The start of the next page (This will always be the previous offset plus the number of rows returned. It does not indicate if there is additional data unless you calculate that the next_offset happens to be closer than it should be.</p> <p>'entry_list' : The records that were retrieved</p> <p>'relationship_list' : The records link field data.</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                 -&gt;GetListEntries(\$sessionID, "Accounts", "",                 "", 0, array("id","name"), array(), "10", 0, false); </pre>

Name	<b>SearchbyModule</b>
Description	<p>Given a list of modules to search and a search string, return the id, module_name, along with the fields</p> <p>We will support Accounts, Bug Tracker, Cases, Contacts, Leads, Opportunities, Project, ProjectTask, Quotes</p>
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$search_string</i> : string to search</p> <p><i>\$modules</i>: array of modules to query</p> <p><i>\$offset</i> : The record offset to start from</p> <p><i>\$max_results</i> : The maximum number of records to return. (default: 10)</p> <p><i>\$assigned_user_id</i>: a user id to filter all records by, leave empty to exclude the filter</p> <p><i>\$select_fields</i> : an array of fields to return. If empty the default return fields will be from the active list view defs</p> <p><i>\$unified_search_only</i>: A boolean indicating if we should only search against those modules participating in the unified search</p> <p><i>\$favorites</i>: A boolean indicating if we should only search against records marked as favorites.</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>return array of return_search_result</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                 -&gt;SearchbyModule(\$sessionID, "A%",                 array("Accounts","Contacts")); </pre>



Name	<b>GetRelationships</b>
Description	retrieve a collection of beans that are related to the specified bean and, optionnally, returns relationship data
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i>: The name of the module that the primary record is from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$module_id</i>: The ID of the bean in the specified module</p> <p><i>\$link_field_name</i>: The name of the link field to return records from. This name should be the name of the relationship.</p> <p><i>\$related_module_query</i>: A portion of the where clause of the SQL statement to find the related items.</p> <p><i>\$related_fields</i>: Array of related bean fields to be returned.</p> <p><i>\$related_module_link_name_to_fields_array</i>: For every related bean returned, specify link fields name to fields info for that bean to be returned.</p> <p><i>\$deleted</i>: false if deleted records should not be include, true if deleted records should be included (default : FALSE)</p> <p><i>\$order_by</i>: SQL order by clause without the phrase 'order by' (default: "")</p> <p><i>\$offset</i> : The record offset to start from</p> <p><i>\$limit</i> : number of results to return. (default: 0 = all)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>'entry_list' -- Array - The records that were retrieved</p> <p>'relationship_list' -- Array - The records link field data.</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                 -&gt;GetRelationships(\$sessionID, "Accounts", "12a27887-f4a0-7f26-74c8-50a47fc41ec9", "contacts", 0, array("id","name"), array(), 0, "name", 0, 0); </pre>

Name	<b>SetRelationship</b>
Description	Sets a single relationship between two SugarBeans
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_name</i>: The name of the module that the primary record is from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$module_id</i>: The ID of the bean in the specified module</p> <p><i>\$link_field_name</i>: name of the link field which relates to the other module for which the relationship needs to be generated.</p> <p><i>\$relatedIDs</i>: array of related record ids for which relationships needs to be generated.</p> <p><i>\$name_value_list</i>: The keys of the array are the SugarBean attributes, the values of the array are the values the attributes should have.</p> <p><i>\$deleted</i>: Optional, if the value 0 or nothing is passed then it will add the relationship for related_ids and if 1 is passed, it will delete this relationship for related_ids (default : FALSE)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>created - integer - How many relationships has been created</p> <p>failed - integer - How many relationship creation failed</p> <p>deleted - integer - How many relationships were deleted</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                 -&gt;SetRelationship(\$sessionID, "Accounts", "12a27887-f4a0-7f26-74c8-50a47fc41ec9", "contacts", array('a7cc315e-bab8-412f-9203-4c752c2ceed7')); </pre>

Name	<b>SetRelationships</b>
Description	Sets multiple relationships between two SugarBeans
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$module_names</i>: Array of the name of the module that the primary record is from. This name should be the name the module was developed under (changing a tab name in studio does not affect the name that should be passed into this method).</p> <p><i>\$module_ids</i>: The array of ID of the bean in the specified module_name</p> <p><i>\$link_field_names</i>: Array of the name of the link field which relates to the other module for which the relationships needs to be generated.</p> <p><i>\$relatedIDs</i>: array of related record ids for which relationships needs to be generated.</p> <p><i>\$name_value_lists</i>: Array of Array. The keys of the inner array are the SugarBean attributes, the values of the inner array are the values the attributes should have.</p> <p><i>\$deleted</i>: Optional, if the value 0 or nothing is passed then it will add the relationship for related_ids and if 1 is passed, it will delete this relationship for related_ids (default : FALSE)</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>created - integer - How many relationships has been created</p> <p>failed - integer - How many relationship creation failed</p> <p>deleted - integer - How many relationships were deleted</p>
Example	

Name	<b>GetQuotePDF</b>
Description	Get the base64 contents of a quote pdf
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$quote_id</i> : ID of quote bean <i>\$pdf_format</i> : name of the output ex. 'Standard' or 'Invoice'
Return	<i>SugarRESTResult</i> object return base64 content pdf file
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser') -&gt;GetQuotePDF(\$sessionID, "40b1b844-f83a-9b74-9888-50a47f5b517c");</pre>

Name	<b>GetQuotePDF</b>
Description	Get the base64 contents of a quote pdf
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$report_id</i> : ID of report bean
Return	<i>SugarRESTResult</i> object return base64 content pdf file
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser') -&gt;GetReportPDF(\$sessionID, "d8e71db6-7fca-2fe7-894d-50a47f58723f");</pre>

Name	<b>GetReportEntries</b>
Description	Retrieve a list of Reports info based on provided IDs.
Parameters	<p><i>\$sessionID</i> : session number returned by call to <i>Connect</i></p> <p><i>\$ids</i>: ID of report bean</p> <p><i>\$select_fields</i>: A list of the fields to be included in the results. This optional parameter allows for only needed fields to be retrieved.</p>
Return	<p><i>SugarRESTResult</i> object</p> <p>'field_list' - Array of Var def information about the returned fields</p> <p>'entry_list' - Array of the records that were retrieved</p>
Example	<pre> \$resultObj = \$this-&gt;get('sugaruser')                -&gt;GetReportEntries(\$sessionID,                array("d8e71db6-7fca-2fe7-894d-50a47f58723f")); </pre>

Name	<b>GetNoteAttachment</b>
Description	Retrieve an attachment from a note
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$note_id</i> : The ID of the appropriate Note
Return	<i>SugarRESTResult</i> object 'id' - The ID of the Note containing the attachment 'filename' - The file name of the attachment 'file' - The binary contents of the file. 'related_module_id' - module id to which this note is related 'related_module_name' - module name to which this note is related
Example	<pre>\$resultObj = \$this-&gt;get('sugaruser')                -&gt;GetNoteAttachment(\$sessionID, "16117d51- c96b-c742-d30b-50a47fe4a9bf");</pre>

Name	<b>SetNoteAttachment</b>
Description	Retrieve an attachment from a note
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$the_note</i> : Array of Note information
Return	<i>SugarRESTResult</i> object 'id' - The ID of the Note containing the attachment
Example	

Name	<b>GetDocumentRevision</b>
Description	This method is used as a result of the .htaccess lock down on the cache directory. It will allow a properly authenticated user to download a document that they have proper rights to download
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$doc_id</i> : The ID of the document
Return	<i>SugarRESTResult</i> object 'id' - The ID of the Note containing the attachment 'document_name' -- name of the document 'revision' - The revision value for this revision 'filename' - The file name of the attachment 'file' - The binary contents of the file.
Example	

Name	<b>SetDocumentRevision</b>
Description	This method is used as a result of the .htaccess lock down on the cache directory. It will allow a properly authenticated user to download a document that they have proper rights to download
Parameters	<i>\$sessionID</i> : session number returned by call to <i>Connect</i> <i>\$document_revisions</i> : Array of document revision information
Return	<i>SugarRESTResult</i> object 'id' - The ID of the Note containing the attachment
Example	

## Result object

---

This object is return by all methods described here.

```
//...  
//...
```

to be done ...