API

- Authentication
- Register
- Departments
- Create a Ticket
- Create a Response
- Auth Required
    - Login
    - Close a Ticket
    - Customer Information
    - Assign a billing ID
    - Grab a Ticket
    - Tickets
    - Responses

Authentication

To use the Ticaga's API you need to submit the Admin email address and API Token with every request as the Authorization: Basic.

Type: POST

```
{
  "result": "true"
}
```

username: The email address of the administrator.

password: The API Token of the administrator.

If these are invalid you will receive an error on every request.

Register

To create a new user account with Ticaga via the API:

URL: https://yourdomain.com/api/register

Type: POST


```
{
  "user": {
    "name": "Michael Dance",
    "email": "dancemike@domain.com",
    "company": "Ticaga Ltd",
    "billing_id": "2",
    "billing_system": "blesta",
    "account_manager": "1",
    "biography": "",
    "updated_at": "2025-07-12T14:14:17.000000Z",
    "created_at": "2025-07-12T14:14:17.000000Z",
    "id": 8
  }
}
```


name: The name of the customer.

email: The email address of the customer.

password: The password of the customer.

company (OPTIONAL): The customer's company name.

billing_id (OPTIONAL): This is the customer's billing system ID, this is used to link them up.

billing_system (OPTIONAL): This is the name of the Billing System which the customer is linked up to.

account_manager (OPTIONAL): This is the ID of the account manager for the customer.

biography (OPTIONAL): This is a little bit about the customer, this is only used at the moment for employee's for the Announcement articles.

updated_at: The date the customer was created on Ticaga. This is automatically set by Ticaga.

created_at: The date the customer was created on Ticaga.This is automatically set by Ticaga.

id: The Ticaga ID of the customer

Departments

This allows you to grab the departments on your Ticaga installation:

URL: https://yourdomain.com/api/departments/public

URL: https://yourdomain.com/api/departments/customers

Type: GET


```
{
  "departments": [
    {
      "id": 2,
      "department_name": "Support",
      "department_description": "Need help, we are here to help.",
      "slug": "support",
      "allows_high_priority": 1,
      "is_public": 2,
      "is_disabled": 0,
      "department_email": "support@example.com",
      "soft_deleted": 0,
      "created_at": "2025-07-19T10:00:08.000000Z",
      "updated_at": null
    }
  ]
}
```

Create a Ticket

URL: https://yourdomain.com/api/tickets/create

Type: POST

```
{
  "id": 8,
}
```

customer_id: The ID of the customer, if the customer doesn't have an ID please set this as 0.
subject: The subject of the ticket.
message: This is the message aka the content of the ticket.
ip_address: This is the customer IP Address.
priority: The priority of the ticket.
department_id: This is the department ID where the ticket will be created and assigned to.
OR instead of the department_id you can send the department_slug
department_slug: This is the department slug where the ticket will be created and assigned to.
organize (OPTIONAL): This is defaulted as "ticket" if no organize is set. To see or change these, please edit the `/app/Enums/Organize.php` file.

The result returns the ticket ID.

Create a Response

URL: https://yourdomain.com/api/responses/create

Type: POST

```
{
  "id": 8,
}
```

ticket_number: This it the ticket ID the response is for.
content: This is the content of the response.
organize (OPTIONAL): This is defaulted as "ticket" if no organize is set. To see or change these, please edit the `/app/Enums/Organize.php` file.
is_note: Is the response a note? 1 for yes or 0 for no.
ip_address: This is the customer IP Address.
employee_response: Is the response from a employee? 1 for yes or 0 for no.
user_id (OPTIONAL): This will submit a user_id for the response, this is handy for Employee replies.

The result returns the ticket ID.

Auth Required

Authorisation via the Connection API call.

Auth Required

Login

To log a customer in:

URL: https://yourdomain.com/api/login

Type: POST


```
{
  "user": {
    "id": 1,
    "name": "Joe Doe",
    "email": "joedoe@example.com",
    "email_verified_at": null,
    "company": "Demo Ltd",
    "phone_number": "",
    "billing_id": "0",
    "billing_system": "",
    "account_manager": "0",
    "biography": "",
    "created_at": "2025-07-13T17:15:25.000000Z",
    "updated_at": "2025-07-13T17:22:37.000000Z"
  }
}
```


email: The email address of the customer.

password: The password of the customer.

Auth Required

Close a Ticket

URL: https://yourdomain.com/api/tickets/close

Type: POST

```
{
  "id": 8,
}
```

customer_id: The ID of the customer, who owns the ticket, this makes sure they can close the ticket.
ticket_id: This is the ticket ID.

The result returns the ticket ID.

Auth Required

Customer Information

To get the customer information from Ticaga:

URL: https://yourdomain.com/api/customers/get/{id}

Type: GET


```
{
  "user": {
    "id": 1,
    "name": "Joe Doe",
    "email": "joedoe@example.com",
    "email_verified_at": null,
    "company": "Demo Ltd",
    "phone_number": "",
    "billing_id": "0",
    "billing_system": "",
    "account_manager": "0",
    "biography": "",
    "created_at": "2025-07-13T17:15:25.000000Z",
    "updated_at": "2025-07-13T17:22:37.000000Z"
  }
}
```


id: Ticaga's customer ID.

If the customer doesn't exist, you get this error:


```
{
  "error": "User not authenticated"
}
```

Auth Required

Assign a billing ID

To assign a billing_id to Ticaga, you need to post their Ticaga ID and their Billing System ID:

URL: https://yourdomain.com/api/customers/assign

Type: POST

```
{
  "user": true
}
```

ticaga_id: Ticaga's customer ID.

billing_id: The billing system customer ID.

billing_system: The billing system name supported by Ticaga Enum.

If the customer doesn't exist, you get this error:

```
{
  "error": "User not authenticated"
}
```

We recommend you store this in the database for the billing system too.

Example from our Blesta plugin:

```
/**
     * Associates/Syncs Blesta Account with Ticaga Account
     *
     * @param int $id The id of the client to fetch
     * @return mixed An stdClass object representing the ticket, or false if none exist
     */
    public function connectAccounts($email_address, $ticaga_id)
    {
□□$ipaddress = $this->get_client_ip_server();
```

```
□□$client_id = $this->Session->read("blesta_client_id");

□□if ($email_address != false)
        {
              $blesta_email = $this->Record->select('email')->from("contacts")-
>where('Client_id', '=', $client_id)->fetch();
              $ticaga_info = json_decode($this->getUserInfoById($ticaga_id), true);

              if($blesta_email->email == $email_address)
              {
                  if($ticaga_info["user"]['id'] == $ticaga_id && $ticaga_info["user"]['email']
== $blesta_email->email)
                  {
                        $company_id = Configure::get('Blesta.company_id');
                        $apiKey = $this->getAPIInfoByCompanyId($company_id)->api_key;
                        $apiURL = $this->getAPIInfoByCompanyId($company_id)->api_url;
                        $apiEmail = $this->getAPIInfoByCompanyId($company_id)->api_email;

□□                  $callvars = array('ticaga_id' => $ticaga_id, "billing_id" => $client_id,
"billing_system" => "Blesta");
                        $this->TicagaSettings->callAPIPost("customers/assign", $callvars,
$apiURL, $apiEmail, $apiKey);

                        $this->Record->duplicate("ticaga_userid", "=", $ticaga_id)-
>insert("ticaga_billing", array('ticaga_userid' => $ticaga_id,'billing_userid' => $client_id,
'email_address' => $email_address, 'billing_system' => 'Blesta'));
                      return true;
                  } else {
                      return false;
                  }
              } else {
                  return false;
              }
        } else {
            return false;
        }
    }
```

Auth Required

Grab a Ticket

How to grab a ticket from Ticaga:

URL: https://yourdomain.com/api/tickets/get/{id}/{ticket_id}

Type: GET


```
{
  "tickets": {
    "id": 2,
    "user_id": 2,
    "subject": "Vitae provident aspernatur minima eum sunt.",
    "message": "Ullam minima est ut ipsa iusto impedit. Dolores quia quos repellat officiis. Autem ad sit qui omnis. Dolorem debitis architecto sit alias laboriosam.\n\nSaepe aut delectus nam omnis eius et. Ratione tempore eum fugiat illum magni quas et nemo. Odit nobis nesciunt voluptate. Excepturi ipsum quam sint in et.",
    "status": "open",
    "priority": "medium",
    "cc": "",
    "assigned": 0,
    "department_id": 2,
    "rating": "0",
    "ip_address": "81.9.172.148",
    "public_hash": "1riUXO8xS9BNZvherjH0",
    "public_name": null,
    "public_email": null,
    "organize": "ticket",
    "date_closed": null,
    "updated_by_client_at": "2025-07-17 18:00:13",
    "first_employee_reply": null,
    "created_at": "2025-07-17T18:00:13.000000Z",
    "updated_at": null,
    "department_name": "Support",
    "department_description": "Need help, we are here to help.",
    "slug": "support",
    "allows_high_priority": 1,
```

```
    "is_public": 2,
    "is_disabled": 0,
    "department_email": "support@example.com",
    "soft_deleted": 0,
    "name": "Prof. Christelle Bechtelar",
    "email": "briana56@example.net",
    "company": "Olson, Hintz and Mueller",
    "company_name": "Olson, Hintz and Mueller",
    "client_name": "Prof. Christelle Bechtelar",
    "ticket_id": 2,
    "department_unique_id": 2,
    "ticket_created": "2025-07-17 18:00:13"
  }
}
```


id: Ticaga's customer ID.

ticket_id: Ticaga's Ticket Number.

If the ticket doesn't belong to the user or doesn't the ticket doesn't exist, you get this error:


```
{
  "error": "No tickets found for this user."
}
```

Auth Required

Tickets

To get all the tickets Ticaga:

URL: https://yourdomain.com/api/tickets/get/all/{ticaga_id}

Type: GET


```
{
  "tickets": [
    {
      "id": 2,
      "user_id": 2,
      "subject": "Vitae provident aspernatur minima eum sunt.",
      "message": "Ullam minima est ut ipsa iusto impedit. Dolores quia quos repellat officiis. Autem ad sit qui omnis. Dolorem debitis architecto sit alias laboriosam.\n\nSaepe aut delectus nam omnis eius et. Ratione tempore eum fugiat illum magni quas et nemo. Odit nobis nesciunt voluptate. Excepturi ipsum quam sint in et.",
      "status": "open",
      "priority": "medium",
      "cc": "",
      "assigned": 0,
      "department_id": 2,
      "rating": "0",
      "ip_address": "81.9.172.148",
      "public_hash": "1riUXO8xS9BNZvherjH0",
      "public_name": null,
      "public_email": null,
      "organize": "ticket",
      "date_closed": null,
      "updated_by_client_at": "2025-07-17 18:00:13",
      "first_employee_reply": null,
      "created_at": "2025-07-17T18:00:13.000000Z",
      "updated_at": null
    }
  ]
```

```
}
```

id: Ticaga's customer ID.

If the customer doesn't have any tickets the array is empty:

```
{
  "tickets": []
}
```

Auth Required

Responses

To get the responses for a ticket in Ticaga:

URL: https://yourdomain.com/api/responses/get/{ticket_id}

Type: GET


```
{
  "responses": [
    {
      "id": 4,
      "user_id": 4,
      "ticket_number": 2,
      "content": "Et animi corrupti ut atque incidunt dolores. Modi odio voluptas corporis explicabo sunt odio. Eveniet quidem voluptas animi atque. Vel quae inventore esse facilis maiores qui harum. Esse enim dolores et et.",
      "organize": "ticket",
      "is_note": 0,
      "employee_response": 1,
      "ip_address": "45.35.88.228",
      "created_at": "2025-07-17T18:00:13.000000Z",
      "updated_at": null,
      "user_name": "Harry Marks",
      "user_email": "charles.renner@example.net"
    },
    {
      "id": 5,
      "user_id": 2,
      "ticket_number": 2,
      "content": "Et temporibus nisi voluptas accusantium provident. Nulla maxime ipsum totam voluptas sunt illo. Nihil minus quod laboriosam placeat qui ad. A fugiat repudiandae quia ab error rem fugiat.",
      "organize": "ticket",
      "is_note": 0,
      "employee_response": 0,
      "ip_address": "42.199.111.145",
```

```
      "created_at": "2025-07-17T18:00:13.000000Z",
      "updated_at": null,
      "user_name": "Prof. Christelle Bechtelar",
      "user_email": "briana56@example.net"
    },
    {
      "id": 6,
      "user_id": 4,
      "ticket_number": 2,
      "content": "Sunt voluptate omnis eum libero et qui. Debitis et est delectus non totam tenetur. Eum rerum aliquam voluptatem ullam reprehenderit nihil dignissimos.",
      "organize": "ticket",
      "is_note": 0,
      "employee_response": 1,
      "ip_address": "148.126.153.97",
      "created_at": "2025-07-17T18:00:13.000000Z",
      "updated_at": null,
      "user_name": "Harry Marks",
      "user_email": "charles.renner@example.net"
    }
  ]
}
```


ticket_id: Ticaga's ticket ID.

If there's no ticket found, you'll get this error:


```
{
  "error": "Ticket not found"
}
```


If there's no responses for the ticket, you'll get this error:


```
{
  "error": "No responses found for this ticket."
}
```