

Responses

Learn about the basics of Graniitti API responses:

- [Response headers](#)
- [Response body](#)
- [Error messages](#)
- [Response status codes](#)

Response headers

The Graniitti API returns the following headers in a response.

Table: Graniitti API response headers

Header	Description	Example value
Access-Control-Allow-Headers	Headers allowed in a request.	Origin, X-Requested-With, Content-Type, Accept
Access-Control-Allow-Methods	HTTP methods allowed for sending a request. The allowed methods vary based on resource and resource type. For more information about which methods the different resource types support, see Reference .	POST, GET, PATCH, DELETE, OPTIONS, PUT
Access-Control-Allow-Origin	Domains from which requests are accepted in cross-origin resource sharing (CORS) .	*
Cache-Control	Caching behavior that client applications and other caching mechanisms along the request /response chain must follow.	no-cache, private
Content-Encoding	Content encoding (compression) applied to the response body.	gzip
Content-Type	Media type of the response body.	application/json
Date	Date and time at which the response was sent.	Wed, 21 Sep 2018 10:00:00 GMT
Server	Information about the server software that handled the request and sent the response.	nginx
Strict-Transport-Security	HTTP Strict Transport Security (HSTS) policy specifying for how long (in seconds) the client application should access the server only through secure HTTPS connections.	max-age=31536000
X-Content-Type-Options	Prevents browsers from MIME-sniffing the response body away from the media type declared in the <code>Content-Type</code> header.	nosniff

X-Frame-Options	Specifies whether the client application is allowed to render the response body in a frame (a <frame>, <iframe>, or <object> element).	SAMEORIGIN
X-XSS-Protection	Specifies whether and what type of cross-site scripting (XSS) protection is enabled for the browser.	1; mode=block

For more information about the standard HTTP headers, see [the header fields registry maintained by IANA](#).

Response body

All responses use the [JSON](#) format for the body.

A single resource is returned as an object. The following example shows a user object.

```
{
  "id": 584,
  "firstname": "User",
  "lastname": "User",
  "email": "user.user@company.com",
  "last_login": null,
  "created": "2016-09-28T06:29:00+00:00",
  "modified": "2016-09-28T06:29:00+00:00",
  "role": "user",
  "permission": "author",
  "notify_via_slack": false,
  "slack_username": null
}
```

A collection of resources is returned as an array of objects. The following example shows an array of user objects.

```
[
  {
    "id": 584,
    "firstname": "User",
    "lastname": "User",
    "email": "user.user@company.com",
    "last_login": null,
    "created": "2016-09-28T06:29:00+00:00",
    "modified": "2016-09-28T06:29:00+00:00",
    "role": "user",
    "permission": "author",
    "notify_via_slack": false,
    "slack_username": null
  },
  {
    "id": 585,
    "firstname": "Power",
    "lastname": "User",
    "email": "power.user@company.com",
    "last_login": null,
    "created": "2016-09-28T06:33:11+00:00",
    "modified": "2016-09-28T06:33:11+00:00",
    "role": "user",
    "permission": "author",
    "notify_via_slack": false,
    "slack_username": null
  }
]
```

If a GET request targets an endpoint that potentially contains multiple resources, the response body is always an array of objects, even in cases where the API only returns a single resource (either because the endpoint only contains a single resource or because filtering resulted in a single resource).

For POST, PUT, and PATCH requests, the response body contains the data for the new or updated resource.

The response body for paginated results looks slightly different. For more information, see [Paginating](#).

Error messages

If a request fails, the response body contains information about the error that occurred, represented as a JSON object.

The following example shows the response to a POST request that tried to create a user that already existed (identified by email address).

```
{
  "code": 0,
  "message": "validation error",
  "errors": {
    "email": [
      "This email address is already in use. Enter a unique email address."
    ]
  }
}
```

Table: Graniitti API error message fields

Error message field	Description
code	Error code. This is always "0".
message	Short error message that provides a summary of the error.
errors	Detailed information about the error, if available.

Response status codes

The API uses standard HTTP status codes to indicate the success or failure of a request. A 2xx status code indicates success, while a 4xx status code indicates failure of some kind.

Table: Graniitti API response status codes

Status code	Description	Solution
200	The server successfully processed the request. The response includes the associated resource data.	
204	The server successfully deleted the target resource. The response does not include any resource data.	
400	The server could not process the request due to an error in the request.	Check that the request uses the correct URL, includes the necessary headers, and, in case of a POST or PUT request, contains a valid JSON body.
401	The server could not authenticate the request because the access token is invalid or missing.	Check that the request includes a valid access token .
403	The server successfully authenticated the request, but could not authorize access to the target resource because the access token does not grant sufficient permission.	Check that you're targeting a resource or resources that you have access to.
404	The server could not find the requested resource.	Check that you're using the correct resource IDs.
412	The server could not process the request because the target resource failed to fulfill one or more preconditions. This status code only applies to DELETE requests to <code>/sites/<site_id>/templates/<template_id></code> .	If you want to delete the content template, first remove it from all modifications.