Xponent Hub Provisioning API 1.0.0

This document describes the Xponent Hub Provisioning API. This API is available for Xponent OEM partners only. The API will retain Xponent nomenclature as it is not rebranded as part of the OEM package.

Use of the Xponent Provisioning API requires a secret access key which is only available to Xponent Owner accounts in an organization. Instructions on how to get the user id and secret to gain a token is provided in the Xponent online documentation and not in this document.

The Content-Type for all request and response bodies should be application/json

- *Note:* All responses include a metadata object, which includes an HTTP status code, message ("OK" for 200 responses, description of error for all others), and, if receiving a 200 response, a numItems property for the length of the array in the response.
- *Note:* All requests are done over HTTPS. HTTP is not accessible.

token ⌄

POST	/token

This endpoint is used to generate an authorization token. If you are an owner of your organization, you can submit your user id, a nonce and the hash of your id and secret in order to receive a token. Tokens have a lifespan of six hours. See online document for hashing examples in multiple languages.

Parameters

Name	Description
Client Credentials *(body)*	Enter your user id, a nonce value and hashed secret key. The nonce value must be a minimum of 30 characters. **Example Value** ```{ "userId": "string", "nonce": "string", "hash": "string" }```

Responses

Code	Description
200	A successful operation returns a token and its expiration time (Unix Epoch time in milliseconds).

Example Value

```
{
  "tokens": [
    {
      "token": "string",
      "expiration": "string"
    }
  ],
  "metadata": {
    "status": 0, "message":
    "string", "numItems": 0
  }
}
```

400	A 400 is a result of one of the following:

1. The following fields are required: userId, nonce, hash
2. The following fields must be non-empty strings: userId, nonce, hash
3. Invalid format for userId
4. Nonce must be a minimum of 30 characters

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401	A 401 is a result of one of the following:

1. Unauthorized - Hash does not match

Example Value

```
{
  "metadata": {
```

```
                "status": 0, "message":
                "string"
              }
            }
```

403   A 403 is a result of one of the following:

1. User with id '{userId}' is not an organization owner
2. User with id '{userId}' does not have a secret key

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404   A 404 is a result of one of the following:

1. User with id '{userId}' does not exist

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

organizations   ⌄

GET   /org

This endpoint returns information about your organization and any child organizations.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name   **Description**

Authorization *** required** An authorization token generated from '/token' route.

string
(header)

> *Note:* Only include the token string. Do not prefix it with any
> specification, such as 'Bearer'.

Responses

Code	Description
200	A successful operation returns an array of organizations. Each organization contains the following: id, name and parentId (parent organization's id). If the parentId for a given organization is listed as null, that entry is the organization of the requesting user.

Example Value

```
{
  "organizations": [
    {
      "id": "string",
      "name": "string",
      "parentId": "string"
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400	A 400 is a result of one of the following:

1. Authorization must be included as a request header

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401	A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

POST   /org

This endpoint allows you to create a new organization (You can only create one organization at a time).

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.

body * required
 	
 (body)
	You are required to pass in a unique name for the organization. The parent is is optional – if it is not provided, the organization's parent will default to the caller's organization.
	
	 Example Value
	
	 ``{``
	 ``  "name": "string",``
	 ``  "parentId": "string"``
	 ``}``

Responses

Code	Description
200	A successful operation returns an array containing the new organization – the new organization will have an id, name and parentId.

Example Value

```
{
  "organizations": [
    {
      "id": "string",
      "name": "string",
      "parentId": "string"
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

| 400 | A 400 is a result of one of the following: |

1. Authorization must be included as a request header
2. Invalid format for parentId
3. Body cannot be empty or null
4. Body must be an object
5. The following fields are required for a new organization: name
5. The following fields, if provided, must be non-empty strings: name, parentId

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

| 401 | A 401 is the result of one of the following: |

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
```

```
          "status": 0, "message":
          "string"
       }
     }
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this parent organization

Example Value

```
{
   "metadata": {
      "status": 0, "message":
      "string"
   }
}
```

404

A 404 is a result of one of the following:

1. Parent organization with id {parentId} not found

Example Value

```
{
   "metadata": {
      "status": 0, "message":
      "string"
   }
}
```

409

A 409 is a result of one of the following:

1. The name {name} is already in use by a different organization

Example Value

```
{
   "metadata": {
      "status": 0, "message":
      "string"
   }
}
```

PUT   **/org/{org_id}**

This endpoint allows you to update an existing organization. You can only update the name of

the organization, and you can only update one organization at a time.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
org_id * required string *(path)*	id of organization to be modified

body * required
(body)
	Both name and parentId are optional, though one or both must exist within the request body
	
	Example Value
	
	```{```
	``` "name": "string",```
	``` "parentId": "string"```
	```}```

Responses

Code	Description

200	A successful operation returns an array containing the updated organization's information including id, name and parentId
	
	Example Value
	
	```{```
	``` "organizations": [```
	``` {```
	``` "id": "string",```
	``` "name": "string",```

```
        "parentId": "string"
      }
    ],
    "metadata": {
      "numItems": 0,
      "status": 0, "message":
      "string"
    }
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id
3. Body cannot be empty or null
4. Body must be an object
5. The following fields, if provided, must be non-empty strings: name

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
```

```
      }
```

404   A 404 is a result of one of the following:

1. Organization with id '{org_id}' not found

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

409   A 409 is a result of one of the following:

1. The name '{name}' is already in use by a different organization

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

DELETE   /org/{org_id}

This endpoint allows you to delete an existing organization (you can only delete one organization at a time)

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

On a successful deletion, this is ***the last time the information about this organization will be accessible.**.*

Parameters

Name	Description

Authorization * required   An authorization token generated from '/token' route.

string
(header)

Note: Only include the token string. Do not prefix it with any specification, such as 'Bearer'.

org_id * required
string
(path)

Organization id to be deleted

Responses

Code	Description
200	A successful operation returns an array containing the deleted organization's information including id, name and parentId.

The organization has been removed from the database and this is ***the last time the data will be accessible***.

Example Value

```
{
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  },
  "organizations": [
    {
      "id": "string",
      "name": "string",
      "parentId": "string"
    }
  ]
}
```

Code	Description
400	A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
```

```
      }
    }
```

401

A 401 is the result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization
2. Not allowed to delete own organization (organization with id '{org_id}')

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404

A 404 is a result of one of the following:

1. Organization with id '{org_id}' not found

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

GET /org/{org_id}/templates

This endpoint allows you to retrieve all templates to which a project within the given organization has access (matches exactly to those templates that would appear in the

Xponent application when adding an item within a project in organization {org_id}).

This includes templates from the organization, and any template from a parent organization that has been marked as global. Any template returned here would be valid for use in the project POST route for org {org_id}.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note*: Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
org_id * required string *(path)*	The organization id for which templates are associated.

Responses

Code	Description

200	A successful operation returns an array of available templates within an organization. Each template contains an id, name and type.
	
	Example Value
	
	<pre>{
	  "templates": [
	    {
	      "name": "string",
	      "id": "string",
	      "type": "string"
	    }
	  ],
	  "metadata": {
	    "numItems": 0,
	    "status": 0, "message":
	    "string"</pre>

```
        }
    }
```

400   A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

401   A 401 is the result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

403   A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

404   A 404 is a result of one of the following:

1. Organization with id '{org_id}' not found

Example Value

```
{
```

```
        "metadata": {
          "status": 0,
          "message": "string"
        }
      }
```

projects ⌄

GET	/project/org/{org_id}

This endpoint allows you to retrieve all projects within a specified organization. The organization must be the caller's organization or a child organization.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
org_id * required string *(path)*	The organization id of the projects you are retrieving.

Responses

Code	Description
200	A successful operation returns an array of projects – each project contains an id, name and a description. **Example Value**

```
{
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  },
  "projects": [
    {
      "name": "string",
      "id": "string",
      "description": "string"
    }
  ]
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

: 1. Invalid user admin permissions for this organization

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
```

```
          }
        }
```

404

A 404 is a result of one of the following:

: 1. Organization with id '{org_id}' not found

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

POST   /project/org/{org_id}

This endpoint allows you to create a new project within a specified organization. The organization to which the project will belong must be the caller's organization or a child organization.

Projects can optionally be populated with the graphs and nodes described in a template (must provide a valid, authorized template id).

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string (header)	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
org_id * required string (path)	The organization id to which the project will belong.
body * required (body)	Name and description are required.

Optionally, you can add a template id to populate the project with its graphs and nodes. Providing a valid, authorized template id this will cause the response to include relevant template connections but those connections must be filled in via the PUT method. This route does not accept template connections, only a template id. ***The response from the POST the only place to get the ids for the template connections for use in the PUT***.

Example Value

```
{
  "name": "string",
  "templateId": "string",
  "description": "string"
}
```

Responses

Code	Description
200	A successful operation returns an array containing the new project – the project will contain an id, name, description and an array of template connections.

These template connections only appear when the templateId is passed and consists of the information needed to set a specific connection within a template. Note: This method does not add any connections to the database. Connections can only be added via the 'PUT' route

Example Value

```
{
  "projects": [
    {
      "id": "string",
      "name": "string", "description":
      "string", "templateConnections": [
        {
          "id": "string",
          "name": "string",
          "type": "string"
        }
      ]
    }
  ],
  "metadata": {
```

```
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id
3. Invalid format for templateId
4. Body cannot empty or null
5. Body must be an object
6. The following fields are required for a new project: name, description
7. The following fields, if provided, must be non-empty strings: name, description, templateId

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization
2. Template with id '{id}' not allowed for use in projects in organization with '{id}'

Example Value

```
{
  "metadata": {
    "status": 0,
```

```
      "message": "string"
    }
  }
```

404   A 404 is a result of one of the following:

               1. Organization with id '{org_id}' not found
               2. Template with id '{id}' not found

               Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

500   A 500 is a result of one of the following:

               1. There was an issue generating from the template provided. Contact your
               organization's administrator.

               Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

PUT   **/project/{project_id}**

This endpoint allows you to update an existing project (you can only modify one project at a
time). You can only modify projects belonging to the your organization or to a child
organization.

This endpoint allows you to (optionally) modify template connections, in addition to modifying
the name and description of the project.

*The response from the POST the only place to get the ids for the template connections
for use in this route.*

You must submit a valid Authorization token retrieved from the /token end point as a
Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
project_id * required string *(path)*	The project id of the project to be updated.
body * required *(body)*	You must include one or multiple of the below fields – see the connection models below to see what belongs in the details section for template connections. **Example Value**

```
{
   "name": "string", "description":
   "string", "connections": [
     {
       "id": "string",
       "details": {
          "type": "string"
       }
     }
   ]
}
```

Responses

Code	Description
200	A successful operation returns an array containing the updated project – the project will contain an id, a name, a description and a connection's object. The connections will contain an array of successfully modified connections and connections that were not able to be be modified will be within 'failures', with a message indicating the reason. **Example Value**

```
{
  "projects": [
    {
      "id": "string",
      "name": "string", "description":
      "string", "connections": {
        "success": [
          {
            "id": "string",
            "name": "string", "message":
            "string"
          }
        ],
        "failure": [
          {
            "id": "string",
            "name": "string", "message":
            "string"
          }
        ]
      }
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for project id
3. Body cannot be empty or null
4. Body must be an object
5. The following fields, if provided, must be non-empty strings: name, description.
6. The following fields are required for each connection: id, details
7. Connections must be an array
8. Invalid format for connection id
9. Connection details must be an object
10. The following fields are required for each connection details: type
11. The following fields must be non-empty strings for all connection details: type

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

```
    }
```

401   A 401 is a result of one of the following:

1. Unauthorized - Token is not valid.

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403   A 403 is a result of one of the following:

1. Invalid user admin permissions for this project

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404   A 404 is a result of one of the following:

1. Project with id '{project_id}' not found.

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

DELETE   /project/{project_id}

This endpoint allows you to delete an existing project (you can only delete one project at a time). The project must belong to the caller's organization or to a child organization.

You must submit a valid Authorization token retrieved from the /token end point as a

Authorization Header field.

On a successful deletion, this is ***the last time the information about this project will be accessible.***

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
project_id * required string *(path)*	The project id of the project to be deleted.

Responses

Code	Description
200	A successful operation returns an array containing the deleted project – the project will contain an id, name, and description. The project has been removed from the database and this is ***the last time the data will be accessible***. **Example Value**

```
{
  "projects": [
    {
      "id": "string",
      "name": "string",
      "organizationId": "string",
      "templateId": "string"
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

```
      }
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for project id

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this project

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404

A 404 is a result of one of the following:

1. Project with id '{project_id}' not found

Example Value

```json
{
  "metadata": {
```

```
            "status": 0,
            "message": "string"
        }
    }
```

users   ⌄

GET	/user

This endpoint returns all users in your organization and all users in child organizations.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.

Responses

Code	Description
200	A successful operation contains information about all users in your organization and users in any child organization. Data includes id, name, email, organizationId, projects and role

Example Value

```
{
    "users": [
        {
            "id": "string",
            "name": "string",
            "organizationId": "string",
```

```
            "email": "string",
            "role": "string",
            "projects": [
              {
                "id": "string",
                "role": "string"
              }
            ]
          }
        ],
        "metadata": {
          "numItems": 0,
          "status": 0, "message":
          "string"
        }
      }
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

POST   /user/org/{org_id}

This endpoint allows owners to add multiple new users to your organization or a child organization.

You must submit a valid Authorization token retrieved from the /token end point as a

Authorization Header field

Parameters

Name	Description

body * required
 	
 (body)
	You must pass an array of user objects containing, at minimum, an email and name.
	
	 A role is optional – if it is not provided it will default user to 'Member' status ('Member' and 'Owner' are only valid values)
	
	 If there is an issue with the creation of any one new user creation, no users will be created until the causing issues are resolved
	
	 Example Value
	
	 <pre>[
	  {
	    "name": "string",
	    "email": "string",
	    "role": "string"
	  }
	]</pre>

org_id * required string *(path)*	The organization id to which to user will be added
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.

Responses

Code	Description
200	A successful operation returns information about the added users - id, name, email, organizationId, projects and role **Example Value**

```json
{
  "users": [
    {
      "id": "string",
      "name": "string",
      "organizationId": "string", "email":
      "string",
      "role": "string",
      "projects": [
        {
          "id": "string",
          "role": "string"
        }
      ]
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for organization id
3. Body cannot be empty or null
4. Body must be an array
5. The following fields are required for all new users: email, name
6. The following fields must be non-empty strings for all users: email, name
7. The following optional fields, if provided, must be non-empty strings for all users: role

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
```

```
        }
    }
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

404

A 404 is a result of one of the following:

1. Organization with id '{org_id}' not found

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

409

The email provided for a new user, '{email}', is already in use by a different accoun

Example Value

```
{
    "metadata": {
        "status": 0, "message":
        "string"
    }
}
```

PUT   **/user/{user_id}**

This endpoint allows you to update an existing user's information, such as name, email and/or role. Only one user can be updated at a time.

You must submit a valid Authorization token retrieved from the /token end point as a

Authorization Header field

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
user_id * required string *(path)*	User id to be updated

body * required
 	
(body)
	User information to be modified – only pass in the values of the fields you want changed
	
	Example Value
	
	``
	{
	  "name": "string",
	  "email": "string",
	  "role": "string"
	}
	``

Responses

Code	Description

200	A successful operation returns information about the updated user - id, name, email, organizationId, projects and role.
	
	Example Value
	
	``
	{
	  "users": [
	    {
	      "id": "string",
	      "name": "string",
	      "organizationId": "string", "email": "string",
	      "role": "string",
	``

```
      "projects": [
        {
          "id": "string",
          "role": "string"
        }
      ]
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for user id
3. Body cannot empty or null
4. Body must be an object
5. The following fields, if provided, must be non-empty strings: name, email, role

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404

A 404 is a result of one of the following:

1. User with id '{user_id}' not found

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

409

A 409 is a result of one of the following:

1. The email provided, '{email}', is already in use by a different account

Example Value

```json
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

DELETE   /user/{user_id}

This endpoint allows you to delete a user from your organization or a child organization (You can only delete one user at a time).

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field.

On a successful deletion, this is *the last time the information about this user will be accessible.*

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
user_id * required string *(path)*	User ID to be deleted

Responses

Code	Description
200	A successful operation returns information about the deleted user - id, name, email, organizationId, projects and role. The user has been removed from the database and this is **the last time the data will be accessible**. **Example Value**

```
{
  "users": [
    {
      "id": "string",
      "name": "string",
      "organizationId": "string", "email":
      "string",
      "role": "string",
      "projects": [
        {
          "id": "string",
          "role": "string"
        }
      ]
    }
  ],
  "metadata": {
    "numItems": 0,
    "status": 0, "message":
    "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for user id

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid user admin permissions for this organization
2. Not allowed to delete self (user with id '{user_id}')

Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

404

A 404 is a result of one of the following:

1. User with id '{user_id}' not found

Example Value

```
{
  "metadata": {
    "status": 0,
```

```
                    "message": "string"
                }
            }
```

GET   /user/{user_id}/loginToken

This endpoint allows you to generate an very short-lived SSO (single sign on) login token for a given user.

You must submit a valid Authorization token retrieved from the /token end point as a Authorization Header field

Parameters

Name	Description
Authorization * required string *(header)*	An authorization token generated from '/token' route. *Note:* Only include the token string. Do not prefix it with any specification, such as 'Bearer'.
user_id * required string *(path)*	User id of the user you wish to authenticate - this is NOT the caller's user id

Responses

Code	Description
| 200 | A successful operation returns an array containing the login token.<br><br>The token will expire after 60 seconds (expiration is Unix Epoch time in milliseconds)<br><br>**Example Value**<br>```
{
  "tokens": [
    {
``` |

```
      "token": "string",
      "expiration": "string"
    }
  ],
  "metadata": { "expiration":
    "string", "status":  0,
    "message": "string"
  }
}
```

400

A 400 is a result of one of the following:

1. Authorization must be included as a request header
2. Invalid format for user id

Example Value

```
{
  "metadata": {
    "status":  0, "message":
    "string"
  }
}
```

401

A 401 is a result of one of the following:

1. Unauthorized - Token is not valid

Example Value

```
{
  "metadata": {
    "status":  0, "message":
    "string"
  }
}
```

403

A 403 is a result of one of the following:

1. Invalid admin permissions for this user

Example Value

```
{
  "metadata": {
    "status":  0, "message":
    "string"
  }
}
```

404   A 404 is a result of one of the following:

             1. User with id '{user_id}' not found

             Example Value

```
{
  "metadata": {
    "status": 0, "message":
    "string"
  }
}
```

Models ⌄

Organization   {
   id: string
   name: string *
   parentId: string
}

Project   {
   id: string
   name: string *
   organizationId: string *
   templateId: string
}

Template_Connections   {
   id: string *
   name: string
   type: string *
                    Enum:

                         [queue, database, soapWebService, webService, twitter, lift, email]
}

DB_Connector   {
   databaseType: string *
                    default: postgresql
                    Enum:

                         [postgresql, mysql, mssql, oracle]

```
    username:       string *
    password:       string *
    hostname:       string *
    port:           integer *
    database:       string *
}


Twitter_Connector  {
    ckey:           string *
    csecret:        string *
    atoken:         string *
    asecret:        string *
}


Rabbit_Queue_Connector  {
    queueType:      string *
                    Enum:

                        [ rabbitmq ]
    readQueue:      string *
    writeQueue:     string *
    username:       string *
    password:       string *
    host:           string *
    ssl:            boolean *
                    default: false
}


Active_Queue_Connector  {
    queueType:      string *
                    Enum:

                        [ activemq ]
    readQueue:      string *
    writeQueue:     string *
    username:       string *
    password:       string *
    host:           string *
    ssl:            boolean *
                    default: false
}


Sqs_Queue_Connector  {
    queueType:          string *
                        Enum:

                            [ sqs ]
    readQueue:          string *
    writeQueue:         string *
    awsAccessKeyId:     string *
    awsSecretAccessKey: string *
    awsRegion:          string *
}
```

```
Soap_Web_Service_Connector     {
    wsdlUrl:          string *
    headerParams:
                      {
                      }
}


Email_Connector   {
    emailType:        string *
                      default:  smtp
                      Enum:

                          Array [ 1 ]
    username:         string *
    password:         string *
    host:             string *
    port:             integer *
}


Web_Service_Connector   {
    endpoint:         string *
    queryParams:
                      {
                      }
    headerParams:
                      {
                      }
}


Lift_Connector    {
    liftEndpoint:     string      *
    accountId:          string    *
    accessKeyId:      string      *
    secretAccessKey: string  *
}


User    {
    id:               string
    name:             string
    organizationId:    string
    email:            string
    role:             string
                      default:  Member
    projects:
                          [       {
                          id:               string
                          role:             string
                      }]
}
```