**Document for Migrating GitLab SaaS to GitLab Self hosted Server**

**Step 1:** Install the GitLab Enterprise Edition with Favour version.

**Step 2:** Getting the UserID from SaaS Gitlab Using Get API

(<Https://gitlab.com/api/v4/groups/:ID/members>).

Example: GET 🡪 <https://gitlab.com/api/v4/groups/62318063/members> with Bearer Token.

Response Body:

{

        "access\_level": 50,

        "created\_at": "2023-01-10T08:43:58.202Z",

        "expires\_at": null,

        "id": 13386739,

        "username": "mohanrajv",

        "name": "mohan raj",

        "state": "active",

        "avatar\_url": "https://secure.gravatar.com/avatar/87971dedc3e579ec95b29d26a7d0bc72?s=80&d=identicon",

        "web\_url": "https://gitlab.com/mohanrajv",

        "membership\_state": "active"

    }

**Step 3:** Getting the User Details Using the UserID API

(<Https://gitlab.com/api/v4/users/:ID>).

Example: GET 🡪 https://gitlab.com/api/v4/users/13386739 with Bearer Token.

Response Body:

{

    "id": 13386739,

    "username": "mohanrajv",

    "name": "mohan raj",

    "state": "active",

    "avatar\_url": "https://secure.gravatar.com/avatar/87971dedc3e579ec95b29d26a7d0bc72?s=80&d=identicon",

    "web\_url": "https://gitlab.com/mohanrajv",

    "created\_at": "2023-01-04T10:28:08.093Z",

    "bio": "",

    "location": "",

    "public\_email": "mohanrajvelu333@gmail.com",

    "skype": "",

    "linkedin": "",

    "twitter": "",

    "website\_url": "",

    "organization": "",

    "job\_title": "",

    "pronouns": "",

    "bot": false,

    "work\_information": null,

    "followers": 0,

    "following": 0,

    "is\_followed": false,

    "local\_time": null

}

**Step 4:** Add User to the Server with respective role using Post API.

**(**Http://<InstanceIp>/api/v4/users**)** with passing the Attribute of email, name, username, password.

Example: POST 🡪 <http://10.100.12.28/api/v4/users> with Bearer Token.

Request Body: It should passed in form-data.

{

“email”: ”mohan@archimedis.io”,  
 “username” : “raj”,  
 “name” : “raj”,  
 “password” : “\*\*\*\*\*\*\*\*”

}

Response Body:

{

    "id": 14,

    "username": "raj",

    "name": "raj",

    "state": "active",

    "avatar\_url": "https://www.gravatar.com/avatar/cecf028ca80e6e91dc7ec0d356e68f6e?s=80&d=identicon",

    "web\_url": "http://gitlab.example.com/raj",

    "created\_at": "2023-01-10T09:49:54.507Z",

    "bio": "",

    "location": null,

    "public\_email": null,

    "skype": "",

    "linkedin": "",

    "twitter": "",

    "website\_url": "",

    "organization": null,

    "job\_title": "",

    "pronouns": null,

    "bot": false,

    "work\_information": null,

    "followers": 0,

    "following": 0,

    "is\_followed": false,

    "local\_time": null,

    "last\_sign\_in\_at": null,

    "confirmed\_at": null,

    "last\_activity\_on": null,

    "email": "mohan@archimedis.io",

    "theme\_id": 1,

    "color\_scheme\_id": 1,

    "projects\_limit": 100000,

    "current\_sign\_in\_at": null,

    "identities": [],

    "can\_create\_group": true,

    "can\_create\_project": true,

    "two\_factor\_enabled": false,

    "external": false,

    "private\_profile": false,

    "commit\_email": "mohan@archimedis.io",

    "shared\_runners\_minutes\_limit": null,

    "extra\_shared\_runners\_minutes\_limit": null,

    "is\_admin": false,

    "note": null,

    "namespace\_id": 43,

    "created\_by": {

        "id": 1,

        "username": "root",

        "name": "Administrator",

        "state": "active",

        "avatar\_url": "https://www.gravatar.com/avatar/e64c7d89f26bd1972efa854d13d7dd61?s=80&d=identicon",

        "web\_url": "http://gitlab.example.com/root"

    },

    "using\_license\_seat": false

}

**Step 5:** Generate the group Export we can use Post API

(<Https://gitlab.com/api/v4/groups/:ID/export>).

Example: POST 🡪 https://gitlab.com/api/v4/groups/62318063/export with Bearer Token.

Response Body:

{

     "message": "202 Accepted"

}

**Step 6:** To Download the group Export using the Get API

(<Https://gitlab.com/api/v4/groups/:ID/export/download>).

Example: GET 🡪 https://gitlab.com/api/v4/groups/62318063/export/download with Bearer Token and hit Send and Download , File Explorer is Opened and Browser the path to a save file with extension of tar.gz

**Step 7:** Import the Group using Post API

(Http://<InstanceIp>/api/v4/groups/import) with passing the Attribute of name, path, and file**.**

Example: POST 🡪 http://10.100.12.28/api/v4/groups/import with Bearer Token.

Request Body: It should be passed in form-data

{

name: SN ,

path : SN,

file : file

}

Response Body:

{

    "message": "202 Accepted"

}

**Step 8:** To find Instance UserID Using Get API

(Http://<InstanceIp>/api/v4/users).

Example: GET 🡪 http://10.100.12.28/api/v4/users with Bearer Token.

**Step 9:** Add the Instance User to the Group with respective Role using Post API

(Http://<InstanceIp>/api/v4/groups/:ID/members) with the Attribute of

Instance User ID and Access level.

Example: POST 🡪 http://10.100.12.28/api/v4/groups/34/members with Bearer Token

Request Body: It should be passed in form-data.

{

user\_id : 12,

access\_level : 40

}

Response Body:

{

     "access\_level": 40,

     "created\_at": "2023-01-10T10:16:22.799Z",

     "created\_by": {

         "id": 1,

         "username": "root",

         "name": "Administrator",

         "state": "active",

         "avatar\_url": "https://www.gravatar.com/avatar/e64c7d89f26bd1972efa854d13d7dd61?s=80&d=identicon",

         "web\_url": "http://gitlab.example.com/root"

     },

     "expires\_at": null,

     "id": 10,

     "username": "support-bot",

     "name": "GitLab Support Bot",

     "state": "active",

     "avatar\_url": "http://gitlab.example.com/uploads/-/system/user/avatar/10/support-bot.png",

     "web\_url": "http://gitlab.example.com/support-bot",

     "membership\_state": "active"

}

**Step 10:** To get the group labels Using Get API

(<Https://gitlab.com/api/v4/groups/:ID/labels>)

Example: GET 🡪 https://gitlab.com/api/v4/groups/34/labels with Bearer Token

Response Body:

{

        "id": 67,

        "name": "Label 1",

        "description": "sn\_test one ",

        "description\_html": "sn\_test one ",

        "text\_color": "#FFFFFF",

        "color": "#00b140",

        "subscribed": false

    }

**Step 11:** Adding the group labels Using Post API

(Http://<InstanceIp>/api/v4/groups/:ID/labels) with Attribute of name, color.

Example: POST 🡪 https://gitlab.com/api/v4/groups/34/labels with Bearer Token

Request Body: It should be passed in form-data.

{

name : Label 3,

color : #FFAABB

}

Response Body:

{

    "id": 78,

    "name": "Label 3",

    "description": null,

    "description\_html": "",

    "text\_color": "#333333",

    "color": "#FFAABB",

    "subscribed": false

}

**Step 12:** To get the group milestones Using Get API

(<Https://gitlab.com/api/v4/groups/:ID/milestones>).

Example: GET 🡪 https://gitlab.com/api/v4/groups/34/milestones with Bearer Token

Response Body:

{

        "id": 10,

        "iid": 1,

        "group\_id": 34,

        "title": "New Milestone 1",

        "description": null,

        "state": "active",

        "created\_at": "2023-01-06T11:51:17.296Z",

        "updated\_at": "2023-01-06T11:51:17.296Z",

        "due\_date": null,

        "start\_date": null,

        "expired": false,

        "web\_url": "http://gitlab.example.com/groups/sn\_group/-/milestones/1"

    }

**Step 13:** Adding the group milestones Using Post API

(Http://<InstanceIp>/api/v4/groups/34/milestones) with Attribute of title.

Example: POST 🡪 https://gitlab.com/api/v4/groups/34/milestones with Bearer Token

Request Body : It should be passed in form-data.

{

title : “New Milestone 3”

}

Response Body :

{

    "id": 12,

    "iid": 2,

    "group\_id": 34,

    "title": "New Milestone 3",

    "description": null,

    "state": "active",

    "created\_at": "2023-01-10T10:29:45.962Z",

    "updated\_at": "2023-01-10T10:29:45.962Z",

    "due\_date": null,

    "start\_date": null,

    "expired": false,

    "web\_url": "http://gitlab.example.com/groups/sn\_group/-/milestones/2"

}

**Step 14:** Add the Integration to the group by manually.

**Step 15:** To generate the project export Using Post API

(<Https://gitlab.com/api/v4/projects/:ID/export>).

Example: POST 🡪 https://gitlab.com/api/v4/projects/42320930/export with Bearer Token

Response Body:

{

    "message": "202 Accepted"

}

**Step 16:** To Download the generated Project Using Get API

(<Https://gitlab.com/api/v4/projects/:ID/export/download>)

Example: GET 🡪 https://gitlab.com/api/v4/projects/42320930/export/download with Bearer Token and hit Send and Download , File Explorer is Opened and Browser the path to a save file with extension of tar.gz

**Step 17:** Import the project using Post API

(Http://<InstanceIp>/api/v4/projects/import) with namespace, name, path, file.

Example: POST 🡪 https://gitlab.com/api/v4/projects/import with Bearer Token

Request Body : It should be passed in form-data.

{

path : “AA”,

name : “AA”,

namespace : “39”,

file : “Select the file”

}

Response Body :

{

    "id": 18,

    "description": null,

    "name": "aa",

    "name\_with\_namespace": "digital / aa",

    "path": "aa",

    "path\_with\_namespace": "digital/aa",

    "created\_at": "2023-01-10T10:49:39.857Z",

    "import\_status": "scheduled",

    "import\_type": "gitlab\_project",

    "correlation\_id": "01GPDMSJVNGPEQGRA6VWP1HB40",

    "failed\_relations": [],

    "import\_error": null,

    "stats": null

}