

Table of Contents

How to use Swagger UI and Postman for BioStar 2 API	1
Swagger UI	1
How to utilize Swagger UI	2
How to utilize Postman with Swagger UI	3

How to use Swagger UI and Postman for BioStar 2 API

Swagger UI

It is a tool which helps test and document ***REST API** easily.

[Request methods for REST API]

Type	Explanation
GET	The GET method requests a representation of the specified resource. Requests using GET should only retrieve data.
POST	The POST method is used to submit an entity to the specified resource.
PUT	The PUT method replaces all current representations of the target resource with the request payload.
DELETE	The DELETE method deletes the specified resource.

We recommend that you should use Local API(after installing BioStar 2 API Server on local PC where BioStar 2 Server is installed). When you send the message to BioStar Server via API, you have to use URL format below:

<http://127.0.0.1:8795/v2/> [url path displayed API reference]

The URL is different depending on BioStar 2 version.

- BioStar v2.3 or below (v1 API) - <http://127.0.0.1:8790/v1/docs/>
- BioStar v2.4 or above (v2 API) - <http://127.0.0.1:8795/v2/docs/>

Public API

- Access Control
- Auth
- Card
- Device
- Device Group
- Door
- Door Group
- Monitoring
- Privilege
- Reference
- Setting
- User
- User Group

BioStar 2 API

HTTP Header 'Content-Language' is used for language. ISO-639-1, ISO-639-3 values are supported.

- Access Control
- Card
- Device Group
- Door
- Door Group
- Device
- Auth
- Monitoring

We provide the information on BioStar 2 API through the Swagger UI so that you can check kinds of APIs and those model schemas before calling API. Below is an example for the way to utilize this tool by using login API.

How to utilize Swagger UI

- **POST /login API**

- 1) Click yellow box about **Model Schema** in **Parameters** section. Then, these contents will be copied and pasted into the **Value** (white box) in the same section.
- 2) Type the values in the white box in order to login to BioStar 2.
- 3) Click **Try it out** button to check the result about response message.

Auth

POST /login

Implementation Notes

Login

Response Class (Status 200)

Model: Model Schema

```
{
  "account_id": "string",
  "email": "string",
  "name": "string",
  "permissions": [
    "id": 0,
    "name": "string",
    "permissions": [
      "allowed_group_id": "string"
    ]
  ]
}
```

Response Content Type: application/json

Parameters

Parameter	Description	Parameter Type	Data Type
user	User	body	Model: Model Schema

Value

```
{
  "mobile_app_version": "string",
  "mobile_device_type": "string",
  "mobile_os_version": "string",
  "notification_token": "string",
  "password": "string",
  "token_id": "string"
}
```

Parameter content type: application/json

Click to set as parameter value

Response Messages

HTTP Status Code	Reason	Response Model
401	Not Authorized	Model: Model Schema
default	Unexpected error	Model: Model Schema

Try it out

How to utilize Postman with Swagger UI

Postman is a Chrome App which helps you confirm the result of calling API easily. We recommend you to use it. By using the postman, you can construct complex HTTP requests quickly, organize them in collections and share them with your co-workers. The biggest advantage is that the history you call API is remained.

You can download the Postman through the Chrome web store.

<https://chrome.google.com/webstore/search/Postman?hl=en>

• POST /login API

In the Swagger UI ..

- 1) Click yellow box about **Model Schema** in **Parameters** section. Then, these contents will be copied and pasted into the **Value** (white box) in the same section.
- 2) Copy all data shown in the white box.

Auth

POST /login

Implementation Notes

Login

Response Class (Status 200)

Model: Model Schema

```
{
  "account_id": "string",
  "email": "string",
  "name": "string",
  "permissions": [
    "id": 1,
    "name": "string",
    "permissions": [
      "allowed_group_id_list": [

```

Response Content Type: application/json

Parameters

Parameter	Type	Description	Parameter Type	Data Type
user	body	User	body	Model: Model Schema

Model: Model Schema

```
{
  "mobile_app_version": "string",
  "mobile_device_type": "string",
  "mobile_os_version": "string",
  "notification_token": "string",
  "password": "string",
  "user_id": "string"
}
```

Click to select parameter value

Response Messages

HTTP Status Code	Reason	Response Model
401	Not Authorized	Model: Model Schema
default	Unexpected error	Model: Model Schema

Model: Model Schema

```
{
  "message": "string",
  "status_code": "string"
}
```

Model: Model Schema

```
{
  "message": "string",
  "status_code": "string"
}
```

Try it out

In the Postman ..

- 3) Select **POST** in the drop box and then enter URL to call login API.
- 4) Select **Body - raw - JSON(application/json)** like below green box.
- 5) Paste the data copied from Swagger UI into the Postman.
- 6) Click **Send** button.
- 7) The response result will be displayed in the bottom of section.

POST https://api.boson2.com/v2/login

Authorization Headers (1) Body Pre-request Script Tests

form-data x-www-form-urlencoded raw binary JSON(application/json)

raw JSON(application/json)

```
{
  "mobile_app_version": "string",
  "mobile_device_type": "string",
  "mobile_os_version": "string",
  "notification_token": "string",
  "password": "string",
  "user_id": "string"
}
```

Body Cookies Headers (1) Tests

Status: 200 OK Time: 2566 ms

Pretty Raw Preview JSON

```
{
  "user_id": "2",
  "login_id": "japan",
  "name": "japan",
  "user_group": {
    "id": "1",
    "name": "All Users"
  },
  "status": "AC",
  "permission": {
    "id": "1",
    "name": "Administrator",
    "users": [
      "permissions": [
        {
          "module": "USER",
          "read": false,
          "write": true,
          "allowed_group_id_list": [
            "1"
          ]
        },
        {
          "module": "CARD",
          "read": false,
          "write": true,
          "allowed_group_id_list": [
            "1"
          ]
        },
        {
          "module": "USER_GROUP",

```

• GET /login API

In the Swagger UI ..

1) Check **Parameter** in **Parameters** section. Both limit and offset are mandatory as URL parameters.

User Show/Hide List Operations Expand Operations

GET /users Get User List

Implementation Notes
Get User List

Response Class (Status 200)
Model **Model Schema**

```
{
  "message": "string",
  "records": [
    {
      "access_groups": {
        {
          "id": 0,
          "included_by_user_group": "string",
          "name": "string"
        }
      }
    }
  ]
}
```

Response Content Type: application/json

With regards to the parameters required, add them with URL
e.g.) <https://api.biostar2.com/v2/users?limit=5&offset=0>

Parameter	Value	Description	Parameter Type	Data Type
group_id	<input type="text"/>	User Group Id	query	long
text	<input type="text"/>	Search text	query	string
limit	<input type="text" value="(required)"/>	Number of results	query	integer
offset	<input type="text" value="(required)"/>	Results data offset	query	integer

Response Messages

HTTP Status Code	Reason	Response Model
401	Not Authorized	Model Model Schema

In the Postman ..

2) Select **GET** in the drop box and then enter URL to call login API. In case of GET message, you should use the necessary parameters in the URL.

Please note that there should be no space in the URL

3) Click **Send** button.

4) The response result will be displayed in the bottom of section.

GET

https://api.biostar2.com/v2/users?limit=5&offset=0

Check the Parameters required!

Params

Send

AuthorizationHeadersBodyPre-request ScriptTests

TypeNo Auth

BodyCookiesHeaders (13)TestsStatus: 200 OK

PrettyRawPreviewJSON

```
1 {
2   "message": "Processed Successfully",
3   "records": [
4     {
5       "access_groups": [],
6       "card_count": 0,
7       "fingerprint_template_count": 0,
8       "last_modify": 0,
9       "name": "Administrator",
10      "password_exist": true,
11      "photo_exist": false,
12      "pin_exist": false,
13      "status": "AC",
14      "user_group": {
15        "id": "1",
16        "name": "All Users"
17      },
18      "user_id": "1"
19    },
20    {
21      "access_groups": [],
22      "card_count": 0,
23      "fingerprint_template_count": 0,
24      "last_modify": 1,
25      "name": "halley",
26      "password_exist": true,
27      "photo_exist": false,
28      "pin_exist": false,
29      "status": "AC",
30      "user_group": {
31        "id": "1",
32        "name": "All Users"
33      },
34      "user_id": "2"
35    },
36    {
37      "access_groups": [],
38      "card_count": 0,
```

From:
<http://kb.supremainc.com/knowledge/> -

Permanent link:
http://kb.supremainc.com/knowledge/doku.php?id=en:how_to_use_swagger_ui_and_postman_for_biostar_2_api&rev=1519691382

Last update: 2018/02/27 09:29