

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	2

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.

When you send the message to BioStar Server via API, you have to use URL format below:

<https://api.biostar2.com/v2/> [url path displayed API reference]

The URL is different depending on BioStar 2 version.

- BioStar v2.3 or below (v1 API) - <https://api.biostar2.com/v1/docs/>
- BioStar v2.4 or above (v2 API) - <https://api.biostar2.com/v2/docs/>

suprema
Public API Quick Start

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.

The screenshot shows the Swagger UI for the `POST /login` API. The interface is divided into several sections:

- Parameters:** A table with columns: Parameter, Value, Description, Parameter Type, and Data Type. The 'user' parameter is listed with a 'body' type and a 'User' data type. A yellow box highlights the 'Value' field for the 'user' parameter, which contains a JSON object:


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

 A red annotation '② Enter the values to login' points to this box.
- Model Schema:** A yellow box highlights the 'Model Schema' for the 'user' parameter, which contains a JSON object:


```
{
  "account_id": "string",
  "email": "string",
  "name": "string",
  "permissions": [
    "id": 0,
    "name": "string",
    "permissions": [
      "allowed_group_id": 0
    ]
  ],
  "role": "string",
  "status": "string",
  "token": "string",
  "user_id": "string",
  "username": "string"
}
```

 A red annotation '① Click here' points to this box.
- Try it out:** A yellow box highlights the 'Try it out' button at the bottom left. A red annotation '③ Click a button to confirm the result of response message' points to this button.

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.

Swagger UI for the **/login** API. The **Parameters** section shows a **user** parameter with a **Model Schema** box containing the following JSON:

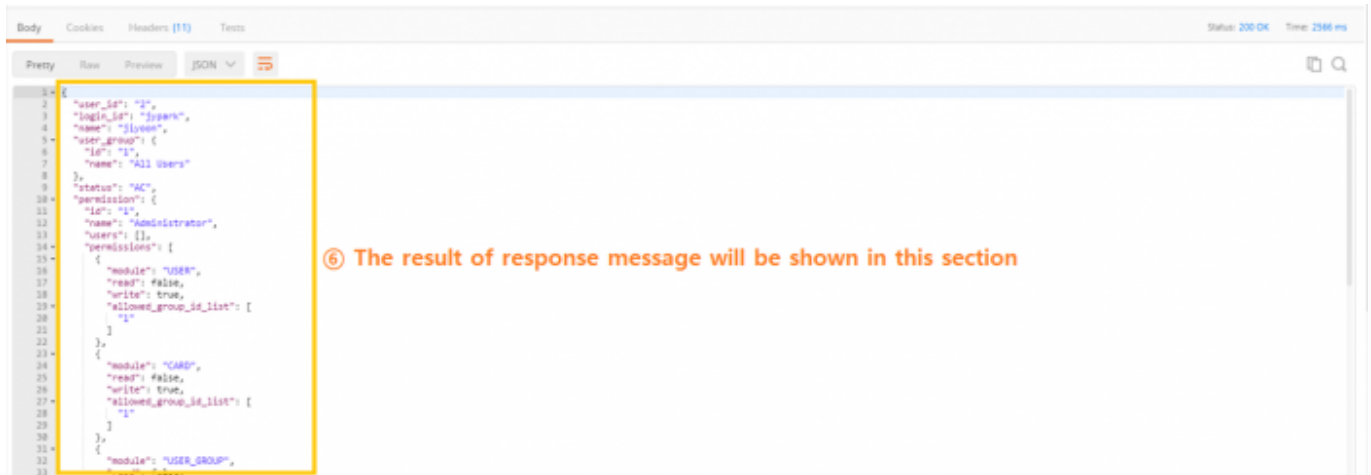
```
{
  "mobile_app_version": "string",
  "mobile_device_type": "string",
  "mobile_os_version": "string",
  "name": "string",
  "notification_token": "string",
  "password": "string",
  "user_id": "string"
}
```

A yellow box highlights the **Model Schema** box with the annotation **1 Click here**. Another yellow box highlights the **Value** text area with the annotation **2 Copy all data shown in the text box**.

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.

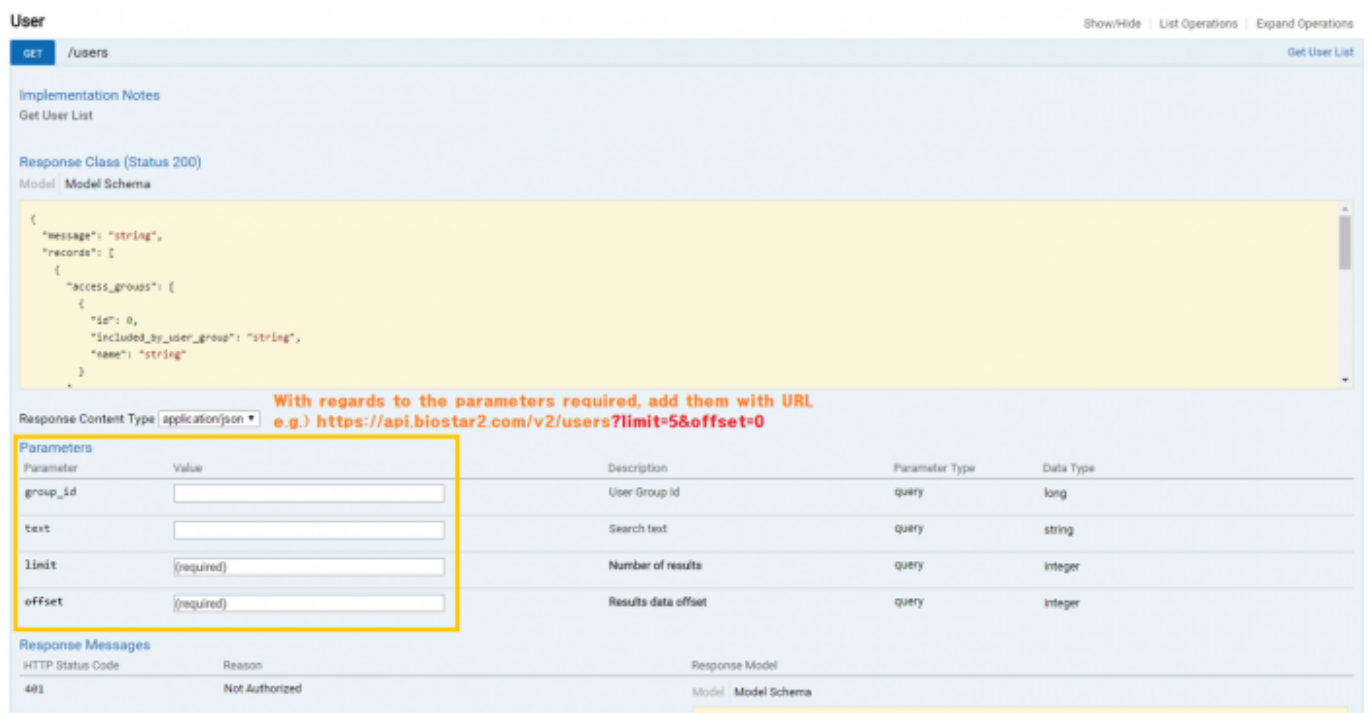
Postman configuration for the **/login** API. The method is set to **POST**, the URL is **https://api.boson2.com/v2/login**, and the body is set to **raw JSON(application/json)**. A yellow box highlights the **Send** button with the annotation **5 Click the "Send" button**. Another yellow box highlights the JSON body with the annotation **4 Enter the values to login**.



• GET /login API

In the Swagger UI ..

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



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=1498546966

Last update: 2017/06/27 16:02