

API Documentation

Users & Groups Project

Version	Date	Author
1.0	30-June-2019	Peyman Ghader Kourehpaz

Index

[1. Login](#)

[Request](#)

[Response](#)

[2. Logout](#)

[Request](#)

[Response](#)

[3. Get Groups](#)

[Request](#)

[Response](#)

[4. Add Group](#)

[Request](#)

[Response](#)

[5. Edit Group \(Get group info \)](#)

[Request](#)

[Response](#)

[6. Update Group](#)

[Request](#)

[Response](#)

[7. Remove Group](#)

[Request](#)

[Response](#)

8. Get Group Users

Request

Response

9. Get Users

Request

Response

10. Add User

Request

Response

11. Edit User (Get User Info with Group Memberships)

Request

Response

12. Update User

Request

Response

13. Remove User

Request

Response

Methods

1. login

Authenticate the user with the system and return access token key , token type with expiration date.

Request

Method	URL
POST	api/auth/login

Type	Params	Values
HEAD	Content-Type	application/json
HEAD	X-Requested-With	XMLHttpRequest
POST	email	String (required)
POST	Password	String (required)
POST	remember_me	Boolean

Authorization

Authorization must be sent with all client requests. Authorization helps the server to validate the request source.

Response

Status	Response
200	Example response: { "access_token": "token", "token_type": "Bearer", "expires_at": "2019-07-07 07:22:22"

	<pre>} Authorization (string) - include (token_type + " " + access_token). all further API calls must have this key in header</pre>
401	<pre>{"error": "Unauthorised"}</pre>

2. logout

Request

Method	URL
ANY -POST -GET	api/auth/logout

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"

Response

Status	Response
success	<pre>{ "message": "Successfully logged out" }</pre>

3. Get Groups

Get groups list.

Request

Method	URL
Get	api/v1/groups

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
GET	page	number

Authorization

The `Authorization` that was given in response to `/api/auth/login`

page

Groups are paginated by 50 , if request has more than 50 groups , then we should specify page value in our requests to paginate them by 50 until we get all groups.

Response

Status	Response
success	<p>Response will be an object containing the list of Groups (array) Paginated by 50 .</p> <p>Example response:</p> <pre>{ "success": 1, "response": { "current_page": 1, "data": [</pre>

	<pre> { "id": 2, "name": "instagram", "created_at": "2019-06-29 14:21:26", "updated_at": "2019-06-30 07:29:22" }, { "id": 3, "name": "facebook", "created_at": "2019-06-29 21:33:50", "updated_at": "2019-06-29 21:33:50" }], "first_page_url": "http://laravelchallenge.test/api/v1/groups?page=1", "from": 1, "last_page": 1, "last_page_url": "http://laravelchallenge.test/api/v1/groups?page=1", "next_page_url": null, "path": "http://laravelchallenge.test/api/v1/groups", "per_page": 50, "prev_page_url": null, "to": 6, "total": 6 } }</pre>
--	---

4. Add Group

Add a new group.

Request

Method	URL
POST	api/v1/add_group

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
Post	group_name	String (required)

Response

Status	Response
success	A Json array Example response: <pre>{ "success": 1, "response": { "name": "My new group", "updated_at": "2019-06-30 08:25:59", "created_at": "2019-06-30 08:25:59", "id": 10 } }</pre>
error	<pre>{"err": 1,"msg": "Error , Wrong Parameters"}</pre>

5. Edit Group

Get Information of a group for editing.

Request

Method	URL
GET	api/v1/edit_group/{id}

ID

Id of the group.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"

Response

Status	Response
success	A json array Example response: <pre>{ "success": 1, "response": { "id": 10, "name": "Group Name", "created_at": "2019-06-30 08:25:59", "updated_at": "2019-06-30 08:25:59" } }</pre>
error	<pre>{"err": 1,"msg": "Group not found"}</pre>

6. Update Group

Update Information of a group for editing.

Request

Method	URL
POST	api/v1/edit_group/{id}

ID

Id of the group.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
post	group_name	String (required)

Response

Status	Response
success	A json array Example response: <pre>{ "success": 1, "response": { "id": 10, "name": "My Updated Group", "created_at": "2019-06-30 08:25:59", "updated_at": "2019-06-30 08:25:59" } }</pre>
error	<pre>{"err": 1,"msg": "Group not found"}</pre>

7. Remove Group

Remove a group.

Request

Method	URL
ANY - POST - GET	api/v1/remove_group/{id}

ID

Id of the group.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"

Response

Status	Response
success	A json array Example response: { "success": 1, "msg": "Group has been removed successfully" }
error	{"err": 2,"msg": "Error : the selected group has members ! we can not remove a group with members ! make it empty and try again."}
error	{"err": 1,"msg": "Group not found"}

8. Get Group Users

Get User members of a group with detail (every users group membership).

Request

Method	URL
GET	api/v1/group_users/{id}

ID

Id of the group.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
GET	page	number

page

Users are paginated by 10 , if request has more than 10 users , then we should specify page value in our requests to paginate them by 10 until we get all users with their group membership relationships for a group.

Response

Status	Response
success	A json array Example response: <pre>{ "success": 1, "response": { "current_page": 1, "data": [{ "id": 10, "name": "payam ghader kourehpaz", "created_at": "2019-06-29 22:09:07",</pre>

	<pre> "updated_at": "2019-06-29 22:09:07", "groups": [{ "id": 2, "name": "instagram", "created_at": "2019-06-29 14:21:26", "updated_at": "2019-06-30 07:29:22", "pivot": { "suser_id": 10, "sgroup_id": 2 } }, { "id": 7, "name": "kurdish boys", "created_at": "2019-06-30 07:29:06", "updated_at": "2019-06-30 07:29:06", "pivot": { "suser_id": 10, "sgroup_id": 7 } }] }, "first_page_url": "http://laravelchallenge.test/api/v1/group_users/2?page=1", "from": 1, "last_page": 1, "last_page_url": "http://laravelchallenge.test/api/v1/group_users/2?page=1", "next_page_url": null, "path": "http://laravelchallenge.test/api/v1/group_users/2", "per_page": 10, "prev_page_url": null, "to": 2, "total": 2 } </pre>
error	<pre> {"err": 1, "msg": "Group not found"} </pre>

9. Get Users

Get User lists with pagination .

Request

Method	URL
ANY -GET -POST	api/v1/users

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
GET	page	number

page

Users are paginated by 10 , if request has more than 10 users , then we should specify page value in our requests to paginate them by 10 until we get all users with their group membership relationships for a group.

Response

Status	Response
success	A json array Example response: <pre>{ "success": 1, "response": { "current_page": 1, "data": [{</pre>

```

        "id": 10,
        "name": "peyman ghader kourehpaz",
        "created_at": "2019-06-29 22:09:07",
        "updated_at": "2019-06-29 22:09:07",
        "groups": [
            {
                "id": 2,
                "name": "instagram",
                "created_at": "2019-06-29 14:21:26",
                "updated_at": "2019-06-30 07:29:22",
                "pivot": {
                    "suser_id": 10,
                    "sgroup_id": 2
                }
            }
        ]
    },
    "first_page_url":
"http://laravelchallenge.test/api/v1/users?page=1",
    "from": 1,
    "last_page": 1,
    "last_page_url":
"http://laravelchallenge.test/api/v1/users?page=1",
    "next_page_url": null,
    "path": "http://laravelchallenge.test/api/v1/users",
    "per_page": 20,
    "prev_page_url": null,
    "to": 3,
    "total": 3
}
}

```

10. Add User

Add a new User. with group memberships .

Request

Method	URL
POST	api/v1/add_user

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
POST	User_name	String (required)
POST	groups.*	Array [integer]

groups.*

An array of specified groups with group value as ID , through this array value parameter , we add user membership to the specified groups through group IDs in the array value .

Example parameter:

groups[] = 2

Groups[] = 3

Here we add User to groups with IDs of 2 and 3 .

Response

Status	Response
success	<p>A Json array</p> <p>Example response:</p> <pre>{ "success": 1, "response": { "name": "new user", "updated_at": "2019-06-30 12:50:40", "created_at": "2019-06-30 12:50:40", "id": 19 } }</pre>

11. Edit User

Get Information of a user for editing with group memberships.

Request

Method	URL
GET	api/v1/edit_user/{id}

ID

Id of the user.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"

Response

Status	Response
success	<p>A json array</p> <p>Example response:</p> <pre>{ "success": 1, "response": { "id": 19, "name": "new user", "created_at": "2019-06-30 12:50:40", "updated_at": "2019-06-30 12:50:40", "groups": [{ "id": 2, "name": "instagram", "created_at": "2019-06-29 14:21:26", "updated_at": "2019-06-30 07:29:22", "pivot": { "suser_id": 19, "sgroup_id": 2 } }] } }</pre>
error	<pre>{"err": 1,"msg": "User not found"}</pre>

12. Update User

Update a user or it's group memberships .

Request

Method	URL
POST	api/v1/edit_user/{id}

ID

Id of the user.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"
POST	User_name	String (required)
POST	groups.*	Array [integer]

groups.*

An array of specified groups with group value as ID , through this array value parameter , we add user membership to the specified groups through group IDs in the array value .

Example parameter:

groups[] = 2

Groups[] = 3

Here we add User to groups with IDs of 2 and 3 .

Response

Status	Response
success	<p>A json array</p> <p>Example response:</p> <pre>{ "success": 1, "response": { "id": 19, "name": "update name and membership", "created_at": "2019-06-30 12:50:40", "updated_at": "2019-06-30 12:59:27", "groups": [{ "id": 2, "name": "instagram", "created_at": "2019-06-29 14:21:26", "updated_at": "2019-06-30 07:29:22", "pivot": { "suser_id": 19, "sgroup_id": 2 } }, { "id": 3, "name": "facebook", "created_at": "2019-06-29 21:33:50", "updated_at": "2019-06-29 21:33:50", "pivot": { "suser_id": 19, "sgroup_id": 3 } }] } }</pre>
error	<pre>{"err": 1,"msg": "User not found"}</pre>

13. Remove User

Remove a User with it's membership relationships.

Request

Method	URL
ANY - POST - GET	api/v1/remove_user/{id}

ID

Id of the user.

Type	Params	Values
HEAD	Authorization	String: "token_type" + " " + "access_token"

Response

Status	Response
success	A json array Example response: { "success": 1, "msg": "User has been removed successfully" }
error	{"err": 1,"msg": "User not found"}