

Group members:

- Yazeed Qholili Arifin (1841720045)
- Abraham Alier (1841720226)
- Ali Abdulsamea Hussen (1841720224)

REST_API

This project uses REST_server which placed inside the REST_API. REST stands for Representational State Transfer. A REST API is a web service which uses HTTP methods likes GET, PUT, POST, DELETE for data manipulation over the cross platforms. To test the API, we will use Postman app in desktop pc or laptop. This app is good to receive JSON values .

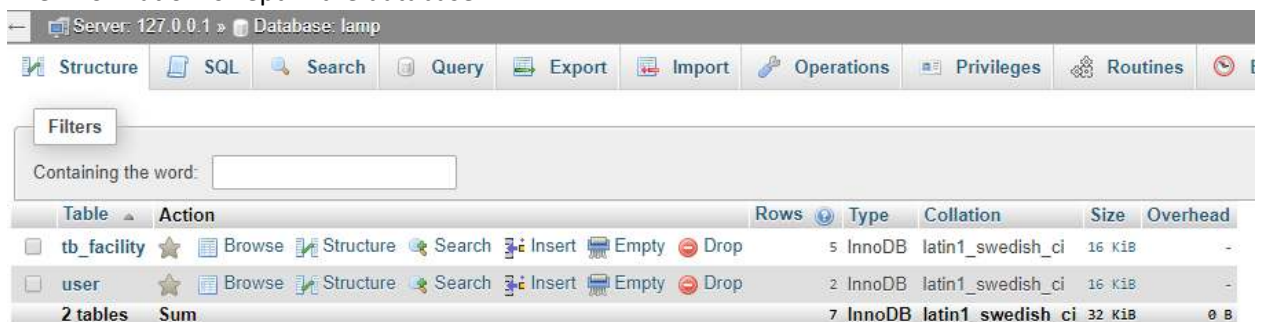
LAMP CONTROLER

Our project, Lamp Controller is an apps used for controlling light. That's for turning off and turning on light.

This application will be controlling light in the following rooms.

- Living room
- Bathroom
- Dining room
- Bedroom
- Kitchen

The information is kept in the database.



The screenshot shows the MySQL Workbench interface for a database named 'lamp' on server '127.0.0.1'. The 'Structure' tab is active, displaying a list of tables. Below the table list is a 'Filters' section with a search box. The table list includes columns for Table, Action, Rows, Type, Collation, Size, and Overhead.

Table	Action	Rows	Type	Collation	Size	Overhead
tb_facility	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 KiB	-
user	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 KiB	-
2 tables	Sum	7	InnoDB	latin1_swedish_ci	32 KiB	0 B

This database has 2 table; tb_facility and user.

Tb_facility contains the following fields.

Options

		fa_living_room	fa_bathroom	fa_dining_room	fa_bedroom	fa_kitchen	id
<input type="checkbox"/>	Edit Copy Delete	1	0	0	0	0	1
<input type="checkbox"/>	Edit Copy Delete	0	0	0	0	0	2
<input type="checkbox"/>	Edit Copy Delete	0	0	0	0	0	3
<input type="checkbox"/>	Edit Copy Delete	0	0	0	0	0	4
<input type="checkbox"/>	Edit Copy Delete	0	1	1	1	1	5

If you click turn off, it would set value of living room to 0

	fa_living_room	fa_bathroom	fa_dining_room	fa_bedroom	fa_kitchen	id
Delete	0	0	0	0	0	1
Delete	0	0	0	0	0	2
Delete	0	0	0	0	0	3
Delete	0	0	0	0	0	4
Delete	0	1	1	1	1	5

Before using this apps, u have to login as either admin or user.

1. Login as admin.

When user login, the valid username and password must be put.

When username and password is wrong, the system would detect and the notice would be shown that the username or password is wrong

Login

Username

admin

Password

.....

Login

Correct username=admin

Password=admin

You have to login as admin with username “admin” and password “admin”. When successful, it would take you to the home page.

Lamp Control

[Home](#) [Add Room](#) [Logout](#)

Room List

1	Delete Room	Control Lamp
2	Delete Room	Control Lamp
3	Delete Room	Control Lamp
4	Delete Room	Control Lamp
5	Delete Room	Control Lamp

If you click control lamp in the 1st column, then you are going to control lamp in the 1st room.

Lamp Controller

Living Room(ON)

Turn On

Turn Off

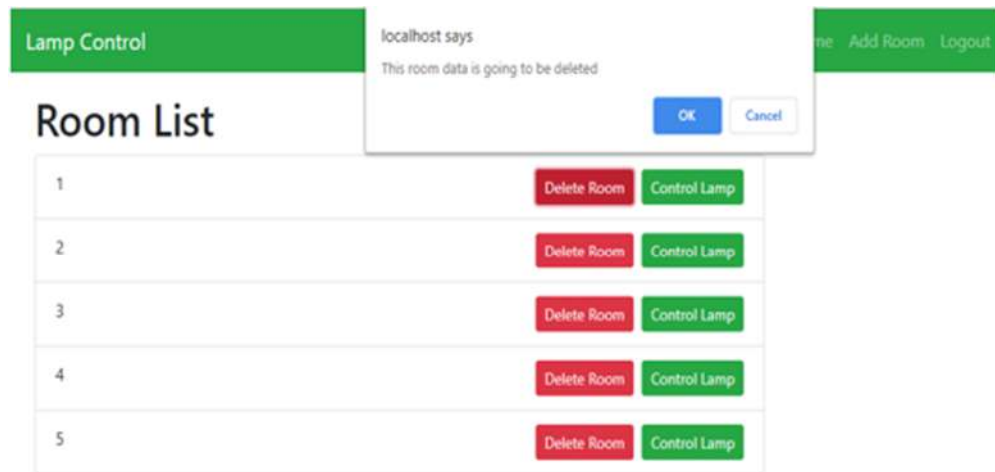
Bath Room(ON)

If you click turn on, it will change the value of from 0 to 1. The opposite thing will happen if you click turn off. The condition of the lamp is represented by a word inside the brackets. In this picture the current lamp state in the living room is (ON), if (OFF) it means the lamp is turned off

	fa_living_room	fa_bathroom	fa_dining_room	fa_bedroom	fa_kitchen	id
Delete	1	0	0	0	0	1
Delete	0	0	0	0	0	2
Delete	0	0	0	0	0	3
Delete	0	0	0	0	0	4
Delete	0	1	1	1	1	5

The opposite thing will happen if you hit the turn off button which will change the value of living room where its ID is matched with user ID. (in this case, our user has ID number 1)

2. DELETE ROOM



If you click ok, then the room data with that specific id number would be deleted.

3. ADD ROOM

Admin can add room in addition to the rooms available in the database. This menu will be activated by clicking button like below.



Then the cardbox will appear

The screenshot shows a form titled 'Input Additional Room ID'. It contains a text input field labeled 'Id' with the value '7' entered. Below the input field is a blue 'Submit' button.

Input Additional Room ID	
Id	7
Submit	

After clicking submit, it would be added.

Data has been Added		×
1	Delete Room	Control Lamp
2	Delete Room	Control Lamp
3	Delete Room	Control Lamp
4	Delete Room	Control Lamp
5	Delete Room	Control Lamp
6	Delete Room	Control Lamp
7	Delete Room	Control Lamp

Check the database to confirm

	fa_living_room	fa_bathroom	fa_dining_room	fa_bedroom	fa_kitchen	id
Copy Delete	0	0	0	0	0	1
Copy Delete	1	0	0	0	0	2
Copy Delete	0	0	0	0	0	3
Copy Delete	0	0	0	0	0	4
Copy Delete	1	1	1	1	1	5
Copy Delete	0	0	0	0	0	6
Copy Delete	0	0	0	0	0	7

The admin can logout. And would be directed to the login page

Masukkan Username dan Password anda

Login

Username

Password

Login

4. Login as user

In order to login as user, you have to put user as username and user as password otherwise you would not be able to login into user page.

Login

Username

user

Password

....

Login

Account: user

Password: user

In the user page, the user is able to:

- i. turn on or turn off lamp.
- ii. logout.

API with REST_Controller

- GET

```
public function index_get()
{
    $id=$this->get('id');
    // $this->db->where('rm_id', $id);
    if ($id===null) {
        $lamp=$this->db->get('tb_facility')->result_array();
        $this->response([
            'status' => true,
            'message' => 'all data',
            'data' => $lamp
        ], REST_Controller::HTTP_OK);
    }
    else{
        $this->db->where('id',$id);
        $lamp=$this->db->get('tb_facility')->result_array();
        $this->response([
            'data' => $lamp
        ], REST_Controller::HTTP_OK);
    }
}

public function getAll($id = null)
{
    if ($id === null) {
        # code...
        return $this->db->get($this->_table)->result_array();
    }else{
        return $this->db->get_where('tb_facility', ['id' => $id])->result_array();
    }
}
```

This is API (in lamp_api.php) which will return the entire rows values by using get method. The postman app will retrieve json values of the script as follows.


```
GET http://localhost/myapp/semester4/jobsheet9/group3/index.php/lamp_api/

Pretty Raw Preview Visualize JSON

1 {
2   "status": true,
3   "message": "all data",
4   "data": [
5     {
6       "fa_living_room": "0",
7       "fa_bathroom": "0",
8       "fa_dining_room": "0",
9       "fa_bedroom": "0",
10      "fa_kitchen": "0",
11      "id": "1"
12    },
13    {
14      "fa_living_room": "0",
15      "fa_bathroom": "0",
16      "fa_dining_room": "0",
17      "fa_bedroom": "0",
18      "fa_kitchen": "0",
19      "id": "2"
20    },
21    {
22      "fa_living_room": "0",
23      "fa_bathroom": "0",
24      "fa_dining_room": "0",
25      "fa_bedroom": "0",
26      "fa_kitchen": "0",
27      "id": "3"
28    }
29  ]
30 }
```

- POST

```
91 public function index_post()
92 {
93     $data = [
94         'id' => $this->input->post('id'),
95         'fa_living_room' => false,
96         'fa_bathroom' => false,
97         'fa_dining_room' => false,
98         'fa_bedroom' => false,
99         'fa_kitchen' => false
100     ];
101
102     if ($this->lamp_model->row_post($data) > 0) {
103         # code...
104         $this->response([
105             'status' => true,
106             'message' => 'new data is inserted',
107             'data' => $data
108         ], REST_Controller::HTTP_CREATED);
109     }else{
110         $this->response([
111             'status' => false,
112             'message' => 'failed to insert new data'
113         ], REST_Controller::HTTP_BAD_REQUEST);
114     }
115 }
```

```

public function row_post($data)
{
    # code...
    $this->db->insert('tb_facility', $data);
    return $this->db->affected_rows();
}

```

The script is used to get insert a new data into the database. The entire rooms are declared with false as default even if we provide certain values into postman to be inserted. False means the lamp is turned off. Picture below is example of JSON values of Postman app.

POST ▼ http://localhost/myapp/semester4/jobsheet9/group3/index.php/lamp_api/ Send ▼

Params ● Authorization Headers (10) ● Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	id	10	
<input type="checkbox"/>	fa_living_room	1	
<input type="checkbox"/>	fa_bathroom	0	
<input type="checkbox"/>	fa_dining_room	0	
<input type="checkbox"/>	fa_bedroom	0	" ...
<input type="checkbox"/>	fa_kitchen	0	
	Key	Value	Description

Pretty Raw Preview Visualize JSON ▼ ↔

```

1  {
2    "status": true,
3    "message": "new data is inserted",
4    "data": {
5      "id": "10",
6      "fa_living_room": false,
7      "fa_bathroom": false,
8      "fa_dining_room": false,
9      "fa_bedroom": false,
10     "fa_kitchen": false
11   }
12 }

```

- PUT

```

public function index_put()
{
    $id=$this->put('id');
    $data = [
        // 'id' => $this->input->put('id'),
        'fa_living_room' => $this->put('fa_living_room'),
        'fa_bathroom' => $this->put('fa_bathroom'),
        'fa_dining_room' => $this->put('fa_dining_room'),
        'fa_bedroom' => $this->put('fa_bedroom'),
        'fa_kitchen' => $this->put('fa_kitchen')
    ];
    if ($this->lamp_model->row_update($data, $id) > 0) {
        $this->response([
            'status' => false,
            'message' => 'failed to update',
            'data' => $data
        ], REST_Controller::HTTP_BAD_REQUEST);
    }else{
        $this->response([
            'status' => true,
            'message' => 'data updated',
            'data' => $data
        ], REST_Controller::HTTP_OK);
    }
}

```

```

public function row_update($data, $id)
{
    $this->db->update('tb_facility', $data, ['id' => $id]);
    return $this->db->affected_rows();
}

```

Put method is used to update values in the table. That's why we must specify the ID that we wanted to modify its value. The following picture is value of JSON in Postman.

PUT ▼ http://localhost/myapp/semester4/jobsheet9/group3/index.php/lamp_api/

Params ● Authorization Headers (10) **Body** ● Pre-request Script Tests Se

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

	KEY	VALUE
<input checked="" type="checkbox"/>	id	10
<input checked="" type="checkbox"/>	fa_living_room	1
<input checked="" type="checkbox"/>	fa_bathroom	0
<input checked="" type="checkbox"/>	fa_dining_room	1
<input checked="" type="checkbox"/>	fa_bedroom	0
<input checked="" type="checkbox"/>	fa_kitchen	1

```
1 {
2   "status": true,
3   "message": "data updated",
4   "data": {
5     "fa_living_room": "1",
6     "fa_bathroom": "0",
7     "fa_dining_room": "1",
8     "fa_bedroom": "0",
9     "fa_kitchen": "1"
10  }
11 }
```

- DELETE

```
public function index_delete()
{
    $id = $this->delete('id');
    if ($id === null) {
        $this->response([
            'status' => false,
            'message' => 'provide an ID'
        ], REST_Controller::HTTP_BAD_REQUEST);
    }else{
        if ($this->lamp_model->row_delete($id)>0) {
            # code...
            echo 'ok';
            $this->response([
                'status' => true,
                'id' => $id,
                'message' => 'deleted'
            ], REST_Controller::HTTP_NO_CONTENT);
        }else{
            echo 'not found';
            $this->response([
                'status' => false,
                'message' => 'id not found!'
            ], REST_Controller::HTTP_BAD_REQUEST);
        }
    }
}

public function row_delete($id){
    // return $this->db->delete($this->_table, array("id"=>$id));
    $this->db->where('id', $id);
    $this->db->delete('tb_facility');
    return $this->db->affected_rows();
    // $this->delete();
}
```

This delete script must have a specified ID to be deleted. The following picture is JSON values of Postman.


DELETE ▼ http://localhost/myapp/semester4/jobsheet9/group3/index.php/lamp_api/

Params ● Authorization Headers (10) **Body ●** Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL

	KEY	VALUE
<input checked="" type="checkbox"/>	id	10
<input type="checkbox"/>	fa_living_room	1
<input type="checkbox"/>	fa_bathroom	0
<input type="checkbox"/>	fa_dining_room	1
<input type="checkbox"/>	fa_bedroom	0
<input type="checkbox"/>	fa_kitchen	1
	Key	Value

Body Cookies (1) Headers (8) Test Results Status: 204 No Content

Pretty Raw Preview Visualize Text ▼ 

1

As we see that there is no content due to REST_Controller::HTTP_NO_CONTENT is executed as the deleted ID