

**LAPORAN PRKTIKUM
PEMROGRAMAN WEB LANJUT
IMPLEMENTASI REST API**



**Untuk mata kuliah :
Pemrograman Web Lanjut**

Disusun Oleh:

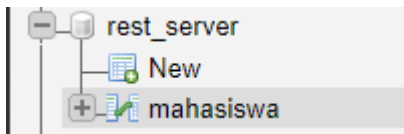
Reza Febriansyah (1841720120)

**JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020**

No	KETERANGAN
1	<p>Configurasi pada config, autoload dan database seperti dibawah ini :</p> <p>pada comfig.php</p> <pre data-bbox="266 459 990 753"> 25 */ 26 \$config['base_url'] = 'http://localhost/rest_server/'; 27 28 /* 29 ----- 30 Index File 31 ----- 32 33 Typically this will be your index.php file, unless you've renamed it to 34 something else. If you are using mod_rewrite to remove the page set this 35 variable so that it is blank. 36 37 */ 38 \$config['index_page'] = ''; 39 </pre> <p>pada autoload.php</p> <pre data-bbox="266 833 1010 1220"> 60 */ 61 \$autoload['libraries'] = array('database'); 62 63 /* 64 ----- 65 \$autoload['helper'] = array('url', 'file'); 66 ----- 67 91 */ 92 \$autoload['helper'] = array('url'); 93 94 /* 95 ----- 96 </pre> <p>pada database.php</p>

```
$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => 'mysql',
    'database' => 'rest_server',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);
```

2 membuat database **rest_server**



yang berisikan table **mahasiswa**

+ Options

	id	nrp	nama	email	jurusan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	043040001	Doddy Ferdiansyah	doy@gmail.com	Teknik Mesin
<input type="checkbox"/> You can also edit most values by double-clicking directly on them.	2	023040123	Erik	erik@gmail.com	Teknik Industri
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	043040321	Rommy Fauzi	rommy@gmail.com	Teknik Planologi
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	033040023	Fajar Darmawan	fajar@yahoo.com	Teknik Informatika
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	113040321	Ferry Mulyanto	ferry@yahoo.com	Manajemen

3 Lakukan copy file di folder **application** dan raplace di folder **Code_kedelapan**,Maka hasilnya akan seperti dibawah ini:

Welcome to CodeIgniter!

[REST Server Tests](#)

The page you are looking at is being generated dynamically by CodeIgniter.

If you would like to edit this page you'll find it located at:

`application/views/welcome_message.php`

The corresponding controller for this page is found at:

`application/controllers/Welcome.php`

If you are exploring CodeIgniter for the very first time, you should start by reading the [User Guide](#).

	maka tahap instalasi berhasil
1	<p>Praktikum Bagian 2 : Membuat Aplikasi CRUD Restful Codeigniter</p> <p>Membuat controller di folder api yang bernama mahasiswa.php</p> <pre> <?php defined('BASEPATH') OR exit('no direct script acces allowed'); require APPPATH . 'libraries/REST_Controller.php'; require APPPATH . 'libraries/Format.php'; class Mahasiswa extends REST_Controller{ public function __construct(){ parent::__construct(); \$this->load->model('Mahasiswa_model', ('mahasiswa')); } public function index_get(){ \$id = \$this->get('id'); if (\$id==null) { \$mahasiswa = \$this->mahasiswa->getMahasiswa(); }else { \$mahasiswa = \$this->mahasiswa->getMahasiswa(\$id); } if (\$mahasiswa) { \$this->response(['status' => ture, 'data' =>\$mahasiswa],REST_Controller :: HTTP_OK); }else{ \$this->response(['status' => false, 'data' =>'id not found'],REST_Controller :: HTTP_NOT_FOUND); } } } </pre>
2	<p>Selanjutnya membuat Mahasiswa_model di folder model</p> <pre> 1 <?php 2 class Mahasiswa extends CI_model { 3 public function get_Mahasiswa(\$id = null){ 4 if (\$id ==null) { 5 return \$this->get('mahasiswa')->result_array(); 6 }else { 7 return \$this->db->get_where('mahasiswa', ['id' => \$id]) -> result_array(); 8 } 9 } 10 } 11 12 </pre>

3

```
1 {
2   "status": true,
3   "data": [
4     {
5       "id": "1",
6       "nrp": "043040001",
7       "nama": "Doddy Ferdiansyah",
8       "email": "doy@gmail.com",
9       "jurusan": "Teknik Mesin"
10    },
11    {
12      "id": "2",
13      "nrp": "023040123",
14      "nama": "Erik",
15      "email": "erik@gmail.com",
16      "jurusan": "Teknik Industri"
17    },
18    {
19      "id": "3",
20      "nrp": "043040321",
21      "nama": "Rommy Fauzi",
22      "email": "rommy@gmail.com",
23      "jurusan": "Teknik Planologi"
24    },
25    {
26      "id": "4",
27      "nrp": "033040023",
28      "nama": "Fajar Darmawan ",
29      "email": "fajar@yahoo.com",
```

Json file yang muncul pada postman.

kita juga dapat melakukan pencarian berdasarkan id

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	id	2
	Key	Value

Body Cookies Headers (7) Test Results

```
1 {
2   "status": true,
3   "data": [
4     {
5       "id": "2",
6       "nrp": "023040123",
7       "nama": "Erik",
8       "email": "erik@gmail.com",
9       "jurusan": "Teknik Industri"
10    }
11  ]
12 }
```

- 4 manambahkan fungsi Create Update dan Delete, maka kira membuar function index_delete pada Mahasiswa.php

```
33     public function index_delete() {
34         $id = $this->delete('id');
35
36         if($id === null) {
37             $this->response([
38                 'status' => false,
39                 'message' => 'provide an id!'
40             ], REST_Controller::HTTP_BAD_REQUEST);
41         }else{
42             if($this->mahasiswa->deleteMahasiswa($id) > 0) {
43                 //ok
44                 $this->response([
45                     'status' => true,
46                     'id' => $id,
47                     'message' => 'deleted.'
48                 ], REST_Controller::HTTP_OK);
49             }else {
50                 // id not found
51                 $this->response([
52                     'status' => false,
53                     'message' => 'id not found!'
54                 ], REST_Controller::HTTP_BAD_REQUEST);
55             }
56         }
57     }
```

- 5
- ```
58 public function index_post() {
59 $data = [
60 'nrp' => $this->post('nrp'),
61 'nama' => $this->post('nama'),
62 'email' => $this->post('email'),
63 'jurusan' => $this->post('jurusan'),
64];
65
66 if($this->mahasiswa->createMahasiswa($data) > 0) {
67 $this->response([
68 'status' => true,
69 'message' => 'New mahasiswa has been created'
70], REST_Controller::HTTP_CREATED);
71 }else {
72 // id not found
73 $this->response([
74 'status' => false,
75 'message' => 'Failed to create new data!'
76], REST_Controller::HTTP_BAD_REQUEST);
77 }
78 }
```

function index\_post

6

```

79 public function index_put() {
80 $id = $this->put('id');
81 $data = [
82 'nrp' => $this->put('nrp'),
83 'nama' => $this->put('nama'),
84 'email' => $this->put('email'),
85 'jurusan' => $this->put('jurusan'),
86];
87
88 if($this->mahasiswa->updateMahasiswa($data, $id) > 0) {
89 $this->response([
90 'status' => true,
91 'message' => 'Data mahasiswa has been updated'
92], REST_Controller::HTTP_CREATED);
93 } else {
94 // id not found
95 $this->response([
96 'status' => false,
97 'message' => 'Failed to update data!'
98], REST_Controller::HTTP_BAD_REQUEST);
99 }
100 }
101 }

```

function index\_put

7

```

1 <?php
2
3 class Mahasiswa_model extends CI_Model {
4 public function getMahasiswa($id = null) {
5 if($id == null) {
6 return $this->db->get('mahasiswa')->result_array();
7 } else {
8 return $this->db->get_where('mahasiswa', ['id' => $id])->result_array();
9 }
10 }
11 public function deleteMahasiswa($id) {
12 $this->db->delete('mahasiswa', ['id' => $id]);
13 return $this->db->affected_rows();
14 }
15 public function createMahasiswa($data) {
16 $this->db->insert('mahasiswa', $data);
17 return $this->db->affected_rows();
18 }
19 public function updateMahasiswa($data, $id) {
20 $this->db->update('mahasiswa', $data, ['id' => $id]);
21 return $this->db->affected_rows();
22 }
23 }

```

membuat tiga function pada Mahasiswa\_modelx

Selanjutnya lakukan operasi **Create** data pada aplikasi postman, untuk operasi **Create, Update** dan **Delete**, arahkan parameter ke **Body** dan **x-www-form-urlencoded** seperti berikut:

The screenshot shows the Postman interface with the 'Body' tab selected. The format is set to 'x-www-form-urlencoded'. A table lists the following key-value pairs:

| KEY     | VALUE |
|---------|-------|
| nrp     |       |
| nama    |       |
| email   |       |
| jurusan |       |

8

The screenshot shows the Postman interface with the 'Body' tab selected. The format is set to 'x-www-form-urlencoded'. A table lists the following key-value pairs:

| KEY     | VALUE              | DESCRIPTION |
|---------|--------------------|-------------|
| nrp     | 0403070028         |             |
| nama    | Joko Anwar         |             |
| email   | jk_anwar@gmail.com |             |
| jurusan | Teknik Kimia       |             |

9

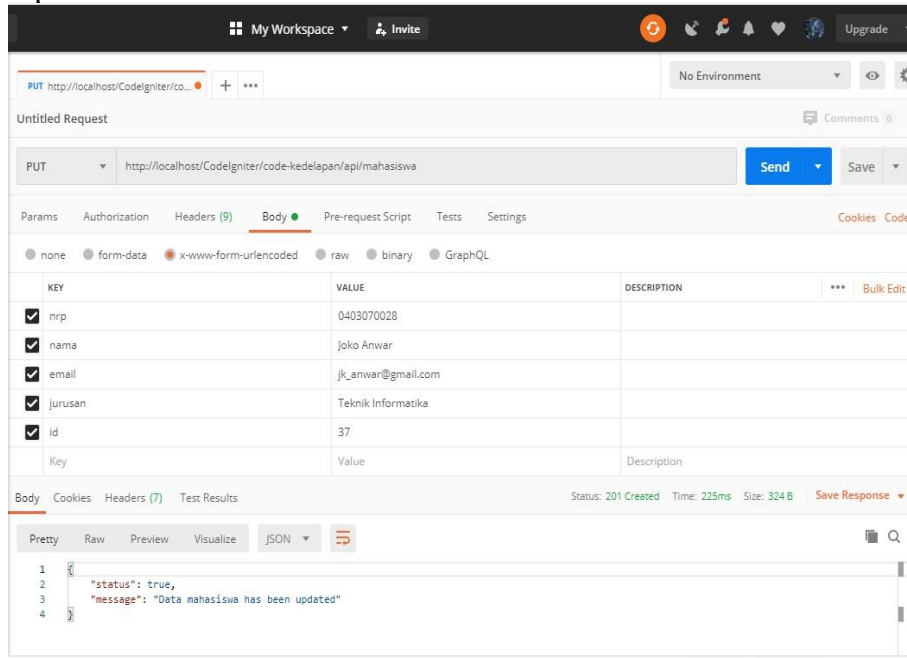
The screenshot shows the Postman interface with a successful POST request to `http://localhost/CodeIgniter/code-kedelapan/api/mahasiswa`. The response is a JSON object:

```
1 {
2 "status": true,
3 "message": "New mahasiswa has been created"
4 }
```

Melakukan Create



- 10 Lakukan juga operasi **Update** data pada aplikasi postman, masukkan parameter id seperti berikut:



- 11 Lakukan juga operasi **Delete** data pada aplikasi postman, parameter yang dimasukkan cukup id saja, seperti berikut:

