

LAPORAN PRAKTIKUM
PEMROGRAMAN WEB LANJUT

Implementasi Restful API



Oleh:

Unero Bhagaskara Ramadhan Marbun

1841720173

TI 2B

TEKNIK INFORMATIKA
TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

Praktikum – Bagian 1: Instalasi library restful codeigniter

The image consists of three numbered screenshots illustrating the setup of a development environment for a Laravel application.

1. Postman Interface: A screenshot of the Postman application window. The interface shows the "My Workspace" tab with a "Launchpad" view. The "Launchpad" view displays a "What's the story, morning glory?" message, a "Start something new" section with options like "Create a request", "Create a collection", and "Create an environment", and a "Recent workspaces" section. The right sidebar shows a "Work smarter with Postman" section with links to "Designing and mocking APIs", "Debugging and manual testing", "Automated testing", "API documentation", "Monitoring", and "Collaboration".

2. Downloading the Laravel Project: A screenshot of a Windows File Explorer window showing the contents of the "Downloads" folder. The folder contains various files and subfolders, including "php-7.3", "phpMyAdmin", "SQLiteManager", "tmp", "www", "abstern", "bank_abstern", "cgi-bin", "CI", "git", "belajar-res", "Code-ked", "Code-keer", "Date modified", "Type", and "Size". The "CI" folder is selected.

3. Localhost/Code-kedelapan: A screenshot of a web browser displaying the "Welcome to CodeIgniter!" page. The page includes a message from the "CodeIgniter" framework, a link to the "User Guide", and a footer indicating the version is "CodeIgniter Version 3.0.2".

4	<p style="text-align: center;">Konfigurasi config.php</p> <pre>\$config['base_url'] = 'http://localhost/code-kedelapan/'; \$config['index_page'] = '';</pre> <p style="text-align: center;">Autoload.php</p> <pre>\$autoload['libraries'] = array('database'); \$autoload['helper'] = array('url');</pre> <p style="text-align: center;">Database.php</p> <pre>\$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,</pre>
5	<p style="text-align: center;">Membuat database rest_server dan datanya</p> <pre>CREATE TABLE `mahasiswa` (`id` int(11) NOT NULL, `nrp` varchar(9) NOT NULL, `nama` varchar(250) NOT NULL, `email` varchar(250) NOT NULL, `jurusan` varchar(64) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1; INSERT INTO `mahasiswa` (`id`, `nrp`, `nama`, `email`, `jurusan`) VALUES (1, '043040001', 'Doddy Ferdiansyah', 'doy@gmail.com', 'Teknik Mesin'), (2, '043040001', 'Erik', 'erik@gmail.com', 'Teknik Industri'), (3, '043040001', 'Rommy Fauzi', 'rommy@gmail.com', 'Teknik Planologi'), (4, '043040001', 'Fajar Darmawan', 'fajar@gmail.com', 'Teknik Informatika'), (5, '043040001', 'Ferry Mulyanto', 'ferry@gmail.com', 'Manajemen'); ALTER TABLE `mahasiswa` ADD PRIMARY KEY (`id`); ALTER TABLE `mahasiswa` MODIFY `id` int(30) NOT NULL AUTO INCREMENT, AUTO INCREMENT=33;</pre>
6	<p style="text-align: center;">Me-replace hasil extract ke project folde application</p>
7	<p style="text-align: center;">Tampilan localhost/code-kedelapan</p> 

Praktikum – Bagian 2: Membuat Aplikasi CRUD Restful Codeigniter

1	<p>Membuat Controller Mahasiswa.php</p> <pre><?php defined('BASEPATH') OR exit('No direct script access allowed'); require APPPATH . '/libraries/REST_Controller.php'; require APPPATH . '/libraries/Format.php'; use Restserver\libraries\REST_Controller; class Mahasiswa extends REST_Controller { function __construct(\$config = 'rest') { parent::__construct(\$config); \$this->load->model('Mahasiswa_model', 'mahasiswa'); } function index_get() { \$id = \$this->get('id'); if (\$id == null) { \$mahasiswa = \$this->mahasiswa->getMahasiswa(); } else { \$this->db->where('id', \$id); \$mahasiswa = \$this->mahasiswa->getMahasiswa(\$id); } if (\$mahasiswa) { \$this->response(['status' => true, 'data' => \$mahasiswa], REST_Controller::HTTP_OK); } else { \$this->response(['status' => true, 'data' => 'id not found'], REST_Controller::HTTP_NOT_FOUND); } } }</pre>
2	<p>Membuat model Mahasiswa_model.php</p> <pre><?php defined('BASEPATH') OR exit('No direct script access allowed'); class Mahasiswa_model extends CI_Model { public function getMahasiswa(\$id=null){ if (\$id == null) { return \$this->db->get('mahasiswa')->result_array(); } else { return \$this->db->get_where('mahasiswa', ['id' => \$id])->result_array(); } } }</pre>
3	<p>Mengecheck API dengan method GET</p> <div><div>GET</div><div>▼</div><div>http://localhost/code-kedelapan/api/Mahasiswa</div></div>

```

    "status": true,
    "data": [
      {
        "id": "1",
        "nrp": "043040001",
        "nama": "Doddy Ferdiansyah",
        "email": "doy@gmail.com",
        "jurusan": "Teknik Mesin"
      },
      {
        "id": "2",
        "nrp": "043040001",
        "nama": "Erik",
        "email": "erik@gmail.com",
        "jurusan": "Teknik Industri"
      },
      {
        "id": "3",
        "nrp": "043040001",
        "nama": "Rommy Fauzi",
        "email": "rommy@gmail.com",
        "jurusan": "Teknik Planologi"
      },
      {
        "id": "4",
        "nrp": "043040001",
        "nama": "Fajar Darmawan",
        "email": "fajar@gmail.com",
        "jurusan": "Teknik Informatika"
      },
      {
        "id": "5",
        "nrp": "043040001",
        "nama": "Ferry Mulyanto",
        "email": "ferry@gmail.com",
        "jurusan": "Manajemen"
      }
    ]
  }
}

```

Berdasarkan Id


GET <http://localhost/code-kedelapan/api/Mahasiswa?id=2>

Params **Authorization** Headers (7) Body Pre-request Script Tests Settings

Query Params

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

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON 

```

1  {
2    "status": true,
3    "data": [
4      {
5        "id": "2",
6        "nrp": "043040001",
7        "nama": "Erik",
8        "email": "erik@gmail.com",
9        "jurusan": "Teknik Industri"
10     }
11   ]
12 }

```

4

Membuat fungsi Create, Update, Delete pada Mahasiswa.php

```
// DELETE
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->mahasiswa->deleteMahasiswa($id) > 0) {
            $this->response([
                'status' => true,
                'id' => $id,
                'message' => 'deleted'
            ], REST_Controller::HTTP_OK);
        } else {
            $this->response([
                'status' => false,
                'message' => 'id not found!'
            ], REST_Controller::HTTP_BAD_REQUEST);
        }
    }
}
```

```
// POST
public function index_post(){
    $data = [
        'nrp' => $this->post('nrp'),
        'nama' => $this->post('nama'),
        'email' => $this->post('email'),
        'jurusan' => $this->post('jurusan'),
    ];

    if ($this->mahasiswa->createMahasiswa($data) > 0) {
        $this->response([
            'status' => true,
            'message' => 'new mahasiswa has been created'
        ], REST_Controller::HTTP_CREATED);
    } else {
        $this->response([
            'status' => false,
            'message' => 'failed to create new data'
        ], REST_Controller::HTTP_BAD_REQUEST);
    }
}
```

```
// PUT
public function index_put(){
    $id = $this->put('id');
    $data = [
        'nrp' => $this->put('nrp'),
        'nama' => $this->put('nama'),
        'email' => $this->put('email'),
        'jurusan' => $this->put('jurusan'),
    ];

    if ($this->mahasiswa->updateMahasiswa($data, $id) > 0) {
        $this->response([
            'status' => true,
            'message' => 'new mahasiswa has been updated'
        ], REST_Controller::HTTP_OK);
    } else {
        $this->response([
            'status' => false,
            'message' => 'failed to update data'
        ], REST_Controller::HTTP_BAD_REQUEST);
    }
}
```

5

Menambahkan kode pada **Mahasiswa_model.php**

```

public function deleteMahasiswa($id){
    $this->db->delete('mahasiswa', ['id' => $id]);
    return $this->db->affected_rows();
}

public function createMahasiswa($data){
    $this->db->insert('mahasiswa', $data);
    return $this->db->affected_rows();
}

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

```

6

Set Body Postman ke **x-www-form-urlencoded**

Params ● Authorization Headers (7) **Body** Pre-request Script Tests Settings

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

Create

POST http://localhost/code-kedelapan/api/Mahasiswa

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

KEY	VALUE
<input checked="" type="checkbox"/> nrp	0403070028
<input checked="" type="checkbox"/> nama	Unero
<input checked="" type="checkbox"/> email	ramadhan@gmail.com
<input checked="" type="checkbox"/> jurusan	Teknik Informatika
Key	Value

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```

1
2  {
3    "status": true,
4    "message": "new mahasiswa has been created"

```

7

Operasi **Update**

PUT http://localhost/code-kedelapan/api/Mahasiswa

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

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

KEY	VALUE
<input checked="" type="checkbox"/> nrp	0403070028
<input checked="" type="checkbox"/> nama	Unero_Ganti
<input checked="" type="checkbox"/> email	ramadhan@gmail.com
<input checked="" type="checkbox"/> jurusan	Teknik Informatika
<input checked="" type="checkbox"/> id	33
Key	Value

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```

1
2  {
3    "status": true,
4    "message": "new mahasiswa has been updated"

```

8

Operasi Delete


DELETE ▼ http://localhost/code-kedelapan/api/Mahasiswa

Params Authorization Headers (9) **Body** ● Pre-request Script Tests Settings

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

KEY	VALUE
<input checked="" type="checkbox"/> id	33
Key	Value

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▼ 

```
1 {
2   "status": true,
3   "id": "33",
4   "message": "deleted"
5 }
```