

# **LAPORAN PRAKTIKUM**

## **Jobsheet-8**

### **Implementasi RestFull API**

#### **Mata Kuliah Pemrograman Web Lanjut**



**Oleh:**

**MINGGAR PUTRA DHEA R.      NIM. 1841720111**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
2020**



### Topik

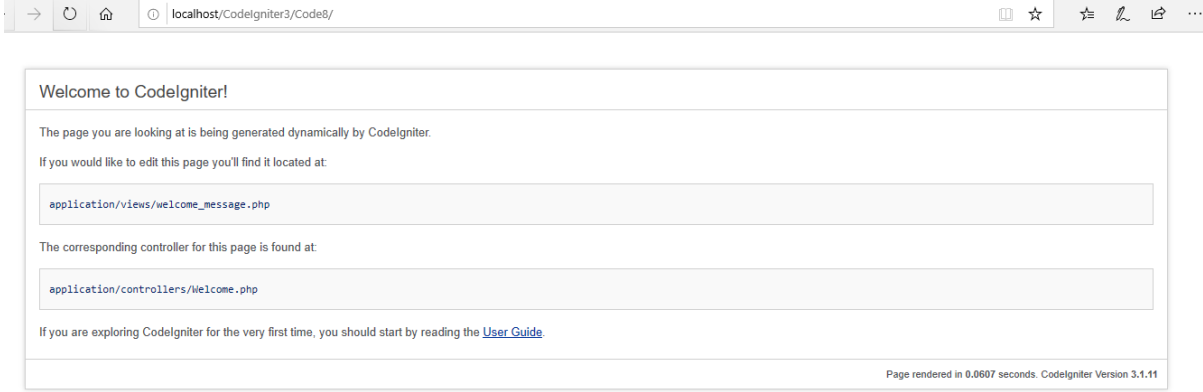
Implementasi Restful API

### Tujuan

Mahasiswa diharapkan dapat:

1. Memahami proses membaca data menggunakan Library Rest Server di Codeigniter.
2. Memahami proses menambah data menggunakan Library Rest Server di Codeigniter.
3. Memahami proses mengubah data menggunakan Library Rest Server di Codeigniter.
4. Memahami proses menghapus data menggunakan Library Rest Server di Codeigniter.

### Praktikum – Bagian 1: Instalasi library restful codeigniter

Langkah	Keterangan
1	Silahkan unduh aplikasi Postman terlebih dahulu.
2	Pada praktikum kali ini kita akan menggunakan source codeigniter yang baru dan Library Rest Server. Untuk versi codeigniter yang digunakan adalah versi 3.1.9. sedangkan untuk Library Rest Server dapat di unduh pada link berikut :
3	<p>Extract codeigniter yang sudah di unduh dan letakkan pada folder htdocs. Beri nama Code-kedelapan dan coba akses di <a href="http://localhost/Code-kedelapan/">http://localhost/Code-kedelapan/</a>. Pastikan tampilan seperti gambar dibawah ini:</p> 
4	Selanjutnya lakukan konfigurasi dasar pada <b>config.php</b> seperti berikut:

```

26 $config['base_url'] = '';
27
28 /*
29 |-----|
30 | Index File
31 |-----|
32 |
33 | Typically this will be your index.php file,
34 | something else. If you are using mod_rewrite
35 | variable so that it is blank.
36 |
37 */
38 $config['index_page'] = 'index.php';
39
40 /*

```

Menjadi seperti berikut :

```

26 $config['base_url'] = 'http://localhost/CodeIgniter3/Code8/';
27
28 /*
29 |-----|
30 | Index File
31 |-----|
32 |
33 | Typically this will be your index.php file, unless you've renamed
34 | something else. If you are using mod_rewrite to remove the page
35 | variable so that it is blank.
36 |
37 */
38 $config['index_page'] = '';
39

```

Dan file **autoload.php**

```

60 /*
61 $autoload['libraries'] = array();
62
63 /*
64 |-----|
65
66
67
68
69
70
71
72 $autoload['helper'] = array();
73
74 /*
75 |-----|
76

```

Menjadi

```

61 $autoload['libraries'] = array('database');
62
63 /*
64 |-----|
65
66
67
68
69
70
71
72 $autoload['helper'] = array('url');
73
74 /*
75 |-----|
76

```

Dan database.php seperti berikut:

```

76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'root',
80     'password' => 'mysql',
81     'database' => 'rest_server',
82     'dbdriver' => 'mysql',
83     'dbprefix' => '',
84     'pconnect' => FALSE,
85     'db_debug' => (ENVIRONMENT !== 'production'),
86     'cache_on' => FALSE,
87     'cachedir' => '',
88     'char_set' => 'utf8',
89     'dbcollat' => 'utf8_general_ci',
90     'swap_pre' => '',
91     'encrypt' => FALSE,
92     'compress' => FALSE,
93     'stricton' => FALSE,
94     'failover' => array(),
95     'save_queries' => TRUE
96 );
97

```

5

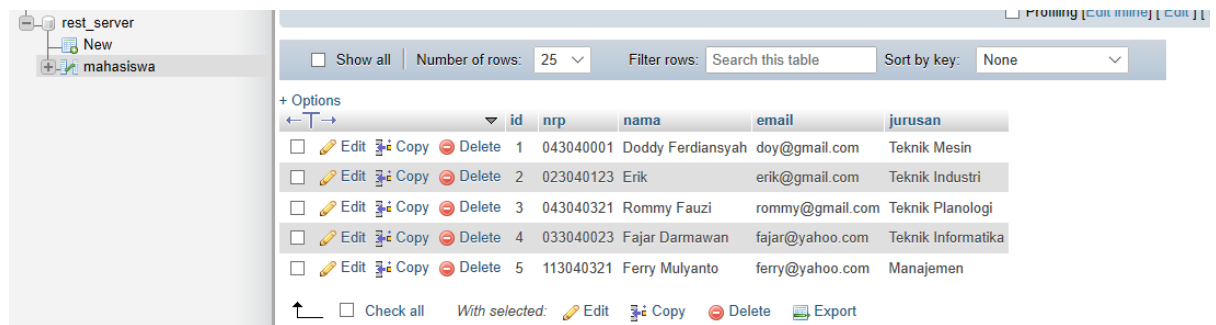
Selanjutnya buatlah database **rest\_server** pada phpmyadmin dan masukkan data dummy. Untuk data dummy dapat dibuat sebagai berikut:

```
CREATE TABLE `mahasiswa` (
  `id` int(11) NOT NULL,
  `nrp` char(9) NOT NULL,
  `nama` varchar(250) NOT NULL,
  `email` varchar(250) DEFAULT 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, '023040123', 'Erik', 'erik@gmail.com', 'Teknik Industri'),
(3, '043040321', 'Rommy Fauzi', 'rommy@gmail.com', 'Teknik Planologi'),
(4, '033040023', 'Fajar Darmawan', 'fajar@yahoo.com', 'Teknik Informatika'),
(5, '113040321', 'Ferry Mulyanto', 'ferry@yahoo.com', 'Manajemen');

ALTER TABLE `mahasiswa`
ADD PRIMARY KEY (`id`);

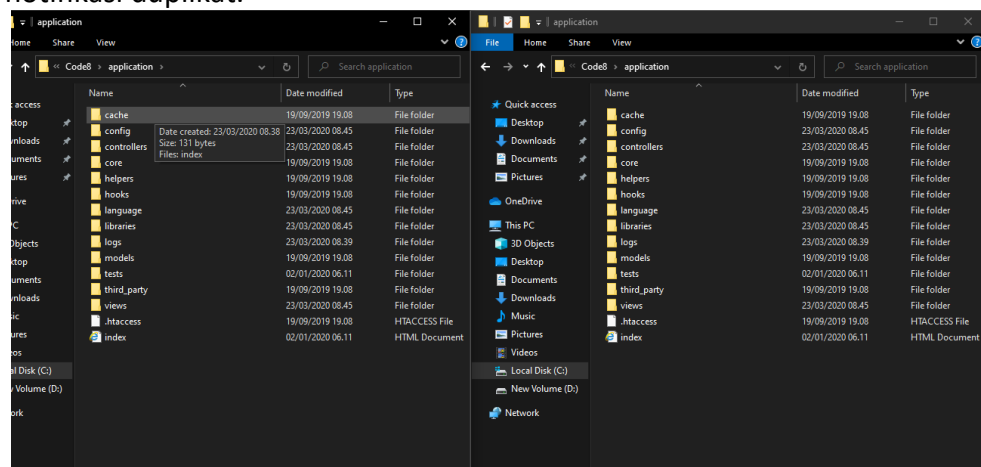
ALTER TABLE `mahasiswa`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=33;
```



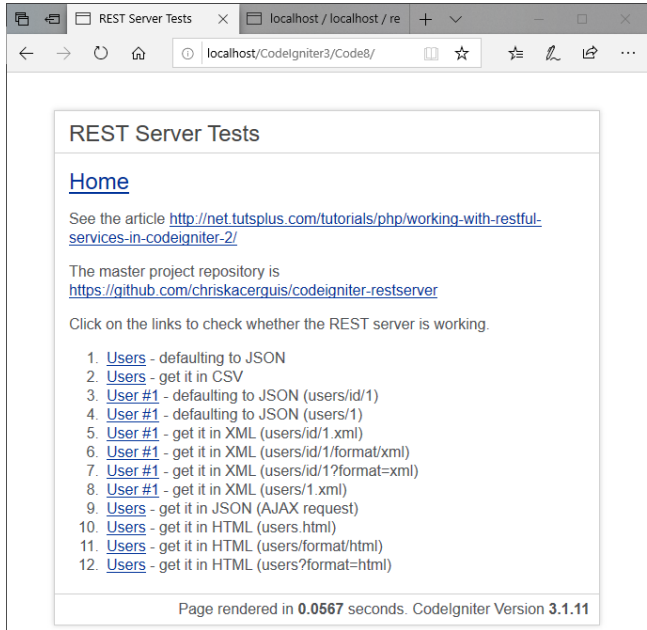
	id	nrp	nama	email	jurusan
<input type="checkbox"/>	1	043040001	Doddy Ferdiansyah	doy@gmail.com	Teknik Mesin
<input type="checkbox"/>	2	023040123	Erik	erik@gmail.com	Teknik Industri
<input type="checkbox"/>	3	043040321	Romy Fauzi	rommy@gmail.com	Teknik Planologi
<input type="checkbox"/>	4	033040023	Fajar Darmawan	fajar@yahoo.com	Teknik Informatika
<input type="checkbox"/>	5	113040321	Ferry Mulyanto	ferry@yahoo.com	Manajemen

6

Selanjutnya extract file Library Rest Server, buka folder **application** dan duplikasi file dan folder tersebut kedalam folder **application** pada project **Code-kedelapan**. Klik **Replace files** jika ada notifikasi duplikat.



Setelah proses duplikasi file selesai, pastikan file-file yang kita sudah konfigurasi di awal tetap terkonfigurasi.

7	<p>Selanjutnya buka project <a href="http://localhost/Code-kedelapan/">http://localhost/Code-kedelapan/</a>. Jika berhasil, tampilan akan berubah menjadi seperti ini:</p> 
8	Tahap instalasi Library Rest Server selesai

## Praktikum – Bagian 2: Membuat Aplikasi CRUD Restful Codeigniter

Langkah	Keterangan
1	<p>Buatlah file <b>Mahasiswa.php</b> pada <b>controller/api</b>. Selanjutnya tuliskan kode berikut:</p>  <pre> 1  application &gt; controllers &gt; api &gt; Mahasiswa.php 2 3  5  require APPPATH . '/libraries/Format.php'; 4 5  6 6 7  7  class Mahasiswa extends REST_Controller{ 8 9  8  function __construct() 10 11  9  { 12 13  10  // Construct the parent class 14 15  11  parent::__construct(); 16 17  12 18  13  \$this-&gt;load-&gt;model('Mahasiswa_model','mahasiswa'); 19 20  14  } 21 22  15 23 24  16  public function index_get(){ 25 26  17  \$id = \$this-&gt;get('id'); 27 28  18  if (\$id == null){ 29 30  19  \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(); 31 32  20  }else{ 33 34  21  \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(\$id); 35 36  22  } 37 38  23 39  24  if(\$mahasiswa){ 40 41  25  \$this-&gt;response([ 42 43  26  'status' =&gt; true, 44 45  27  'data' =&gt; \$mahasiswa 46 47  28  ], REST_Controller::HTTP_OK); 48 49  29  }else{ 50 51  30  \$this-&gt;response([ 52 53  31  'status' =&gt; FALSE, 54 55  32  'message' =&gt; 'id not found' 56 57  33  ], REST_Controller::HTTP_NOT_FOUND); 58 59  34  } 60 61  35  } 62 63  36  } 64 65  37  } 66 67  38  } 68 69  39  } 70 71  40  } 72 73  41  } 74 75  42  } 76 77  43  } 78 79  44  } 80 81  45  } 82 83  46  } 84 85  47  } 86 87  48  } 88 89  49  } 90 91  50  } 92 93  51  } 94 95  52  } 96 97  53  } 98 99  54  } 100 101  55  } 102 103  56  } 104 105  57  } 106 107  58  } 108 109  59  } 110 111  60  } 112 113  61  } 114 115  62  } 116 117  63  } 118 119  64  } 120 121  65  } 122 123  66  } 124 125  67  } 126 127  68  } 128 129  69  } 130 131  70  } 132 133  71  } 134 135  72  } 136 137  73  } 138 139  74  } 140 141  75  } 142 143  76  } 144 145  77  } 146 147  78  } 148 149  79  } 150 151  80  } 152 153  81  } 154 155  82  } 156 157  83  } 158 159  84  } 160 161  85  } 162 163  86  } 164 165  87  } 166 167  88  } 168 169  89  } 170 171  90  } 172 173  91  } 174 175  92  } 176 177  93  } 178 179  94  } 180 181  95  } 182 183  96  } 184 185  97  } 186 187  98  } 188 189  99  } 190 191  100  } 192 193  101  } 194 195  102  } 196 197  103  } 198 199  104  } 200 201  105  } 202 203  106  } 204 205  107  } 206 207  108  } 208 209  109  } 210 211  110  } 212 213  111  } 214 215  112  } 216 217  113  } 218 219  114  } 220 221  115  } 222 223  116  } 224 225  117  } 226 227  118  } 228 229  119  } 230 231  120  } 232 233  121  } 234 235  122  } 236 237  123  } 238 239  124  } 240 241  125  } 242 243  126  } 244 245  127  } 246 247  128  } 248 249  129  } 250 251  130  } 252 253  131  } 254 255  132  } 256 257  133  } 258 259  134  } 260 261  135  } 262 263  136  } 264 265  137  } 266 267  138  } 268 269  139  } 270 271  140  } 272 273  141  } 274 275  142  } 276 277  143  } 278 279  144  } 280 281  145  } 282 283  146  } 284 285  147  } 286 287  148  } 288 289  149  } 290 291  150  } 292 293  151  } 294 295  152  } 296 297  153  } 298 299  154  } 300 301  155  } 302 303  156  } 304 305  157  } 306 307  158  } 308 309  159  } 309 </pre>

2

Selanjutnya Buatlah file **Mahasiswa\_model.php** pada model. Selanjutnya tuliskan kode berikut:

```
application > models > Mahasiswa_model.php
1  <?php
2  class Mahasiswa_model extends CI_model{
3      public function getMahasiswa($id = null){
4          if ($id == null){
5              return $this->db->get('mahasiswa')->result_array();
6          }else{
7              return $this->db->get_where('mahasiswa', ['id' => $id])->result_array();
8          }
9      }
10 }
11 ?>
```

3

Buka aplikasi postman, masukkan alamat <http://localhost/Code-kedelapan/api/mahasiswa> dengan metode **GET**. dan jika berhasil maka tampilannya akan seperti ini:

GET localhost/CodeIgniter3/Code8/api/mahasiswa

Status: 200 OK Time: 167ms Size: 805 B

```
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",
```

Kita juga dapat melakukan pencarian berdasarkan id, hasilnya akan seperti ini:

GET localhost/CodeIgniter3/Code8/api/mahasiswa?id=2

Status: 200 OK Time: 148ms Size: 383 B

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

4

Untuk fungsi lain **Create, Update** dan **Delete**. Lanjutkan kode berikut pada file **mahasiswa.php**:

```

39     function index_delete() {
40         $id = $this->delete('id');
41
42         if ($id === null) {
43             $this->response([
44                 'status' => false,
45                 'message' => 'provide an id !'
46             ], REST_Controller::HTTP_BAD_REQUEST);
47         } else {
48             if ($this->mahasiswa->deleteMahasiswa($id) > 0) {
49                 $this->response([
50                     'status' => true,
51                     'id' => $id,
52                     'message' => 'deleted.'
53                 ], REST_Controller::HTTP_OK);
54             } else {
55                 $this->response([
56                     'status' => false,
57                     'message' => 'id not found!'
58                 ], REST_Controller::HTTP_BAD_REQUEST);
59             }
60         }
61     }
62
63
64     public function index_post(){
65         $data = [
66             'nrp' => $this->post('nrp'),
67             'nama' => $this->post('nama'),
68             'email' => $this->post('email'),
69             'jurusan' => $this->post('jurusan'),
70         ];
71
72         if ($this->mahasiswa->createMahasiswa($data) > 0) {
73             $this->response([
74                 'status' => true,
75                 'message' => 'new mahasiswa has been created'
76             ], REST_Controller::HTTP_CREATED);
77         } else {
78             $this->response([
79                 'status' => false,
80                 'message' => 'failed to create new data!'
81             ], REST_Controller::HTTP_BAD_REQUEST);
82         }
83     }
84
85
86     public function index_put(){
87         $id = $this->put('id');
88         $data = [
89             'nrp' => $this->put('nrp'),
90             'nama' => $this->put('nama'),
91             'email' => $this->put('email'),
92             'jurusan' => $this->put('jurusan'),
93         ];
94         if ($this->mahasiswa->updateMahasiswa($data, $id) > 0) {
95             $this->response([
96                 'status' => true,
97                 'message' => 'new mahasiswa has been updated'
98             ], REST_Controller::HTTP_OK);
99         } else {
100             $this->response([
101                 'status' => false,
102                 'message' => 'failed to update data!'
103             ], REST_Controller::HTTP_BAD_REQUEST);
104         }
105     }
106 }
107 ?>

```

5

Dan lanjutkan kode pada file **Mahasiswa\_model.php**:

```

15     public function deleteMahasiswa($id){
16         $this->db->delete('mahasiswa', ['id' => $id]);
17         return $this->db->affected_rows();
18     }
19
20
21     public function createMahasiswa($data){
22         $this->db->insert('mahasiswa', $data);
23         return $this->db->affected_rows();
24     }
25
26
27     public function updateMahasiswa($data, $id){
28         $this->db->update('mahasiswa', $data, ['id' => $id]);
29         return $this->db->affected_rows();
30     }
31
32
33 }
34
35 /* End of file Mahasiswa_model.php */
36
37 ?>

```

6

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:

Untitled Request

POST localhost/CodeIgniter3/Code8/api/mahasiswa?id=2

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

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

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> nrp	0403070028	
<input checked="" type="checkbox"/> nama	Joko Anwar	
<input checked="" type="checkbox"/> email	jk_anwar@gmail.com	
<input checked="" type="checkbox"/> jurusan	Teknik Kimia	
Key	Value	Description

Body Cookies Headers (7) Test Results Status: 201 Created Time: 178ms Size: 325 B Save Response

Pretty Raw Preview Visualize JSON

```

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

```

7

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

PUT localhost/CodeIgniter3/Code8/api/mahasiswa?id=33

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

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> nrp	0403070028	
<input checked="" type="checkbox"/> nama	Joko Anwar	
<input checked="" type="checkbox"/> email	jk_anwar@gmail.com	
<input checked="" type="checkbox"/> jurusan	Teknik Informatika	
<input checked="" type="checkbox"/> id	33	
Key	Value	Description

Body Cookies Headers (7) Test Results Status: 200 OK Time: 145ms Size: 320 B Save Response

Pretty Raw Preview Visualize JSON

```

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

```



8

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

