

# **LAPORAN PRAKTIKUM PEMROGRAMAN WEBSITE LANJUT**

## **RESTFUL API**



**Oleh :**

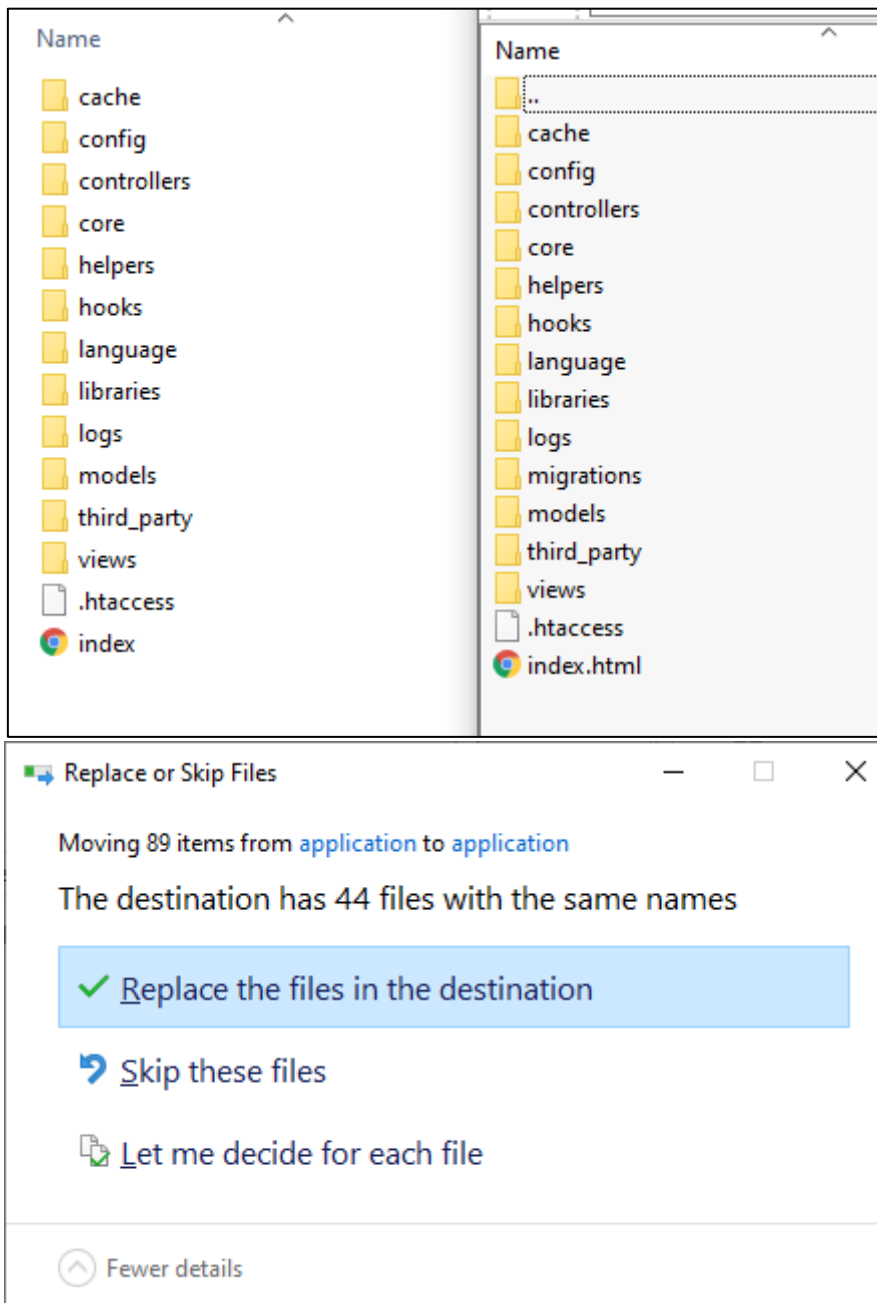
**A.SAFA DHIATA / NIM : 1941723012  
KELAS TI 2B**

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

## Praktikum – Bagian 1: Instalasi library restful codeigniter

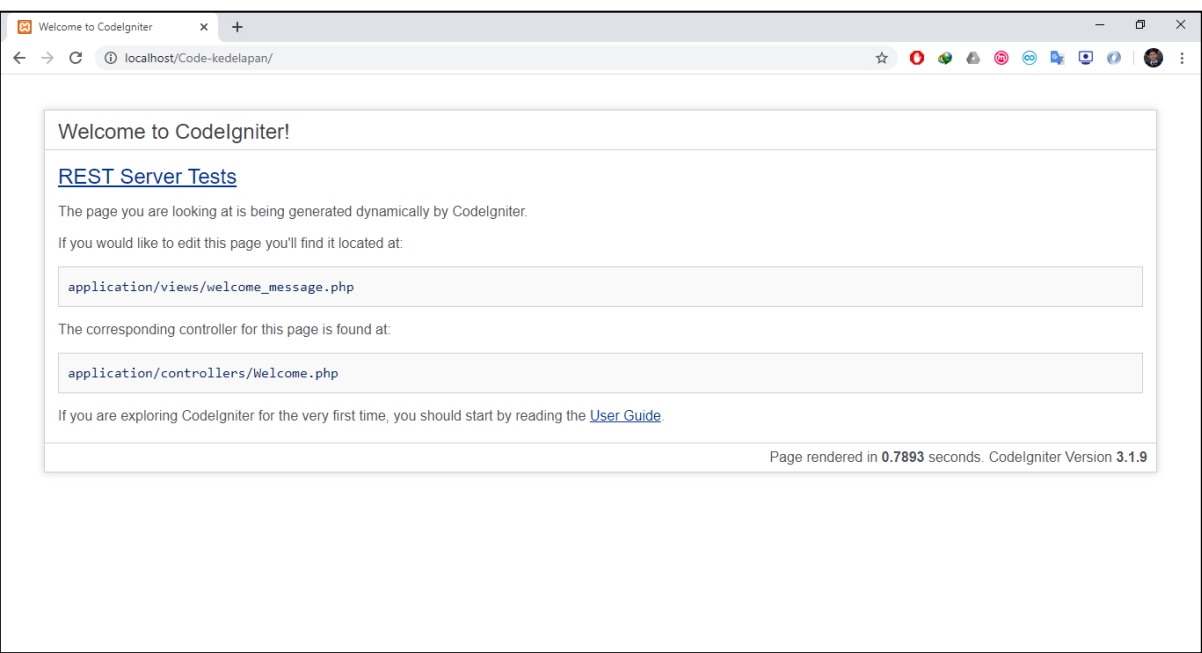
Langkah	Keterangan
1	Silahkan unduh aplikasi Postman terlebih dahulu.
2	Pada praktikum kali ini kita akan menggunakan source codeigniter yang baru dan Library Rest Server. Untuk versi codeigniter yang digunakan adalah versi 3.1.9. sedangkan untuk Library Rest Server dapat di unduh pada link berikut :
3	<p>Extract codeigniter yang sudah di unduh dan letakkan pada folder httdocs. 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	<p>Selanjutnya lakukan konfigurasi dasar pada <b>config.php</b> seperti berikut:</p> <pre>26 \$config['base_url'] = 'http://localhost/Code-kedelapan/';  27 \$config['index_page'] = ''; 28 \$config['uri_protocol'] = 'REQUEST_URI'; 29</pre> <p>Dan file <b>autoload.php</b></p> <pre>61 \$autoload['libraries'] = array('database'); 62 \$autoload['drivers'] = array();  63 \$autoload['helper'] = array('url'); 64</pre> <p>Dan database.php seperti berikut:</p>

	<pre> 75 76 \$db['default'] = array( 77     'dsn' =&gt; '', 78     'hostname' =&gt; 'localhost', 79     'username' =&gt; 'root', 80     'password' =&gt; '', 81     'database' =&gt; 'rest_server', 82     'dbdriver' =&gt; 'mysqli', 83     'dbprefix' =&gt; '', </pre>
5	<p>Selanjutnya buatlah database <b>rest_server</b> pada phpmyadmin dan masukkan data dummy. Untuk data dummy dapat dibuat sebagai berikut:</p> <pre> 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; </pre>
6	<p>Selanjutnya extract file Library Rest Server, buka folder <b>application</b> dan duplikasi file dan folder tersebut kedalam folder <b>application</b> pada project <b>Code-kedelapan</b>. Klik <b>Replace files</b> jika ada notifikasi duplikat.</p>



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

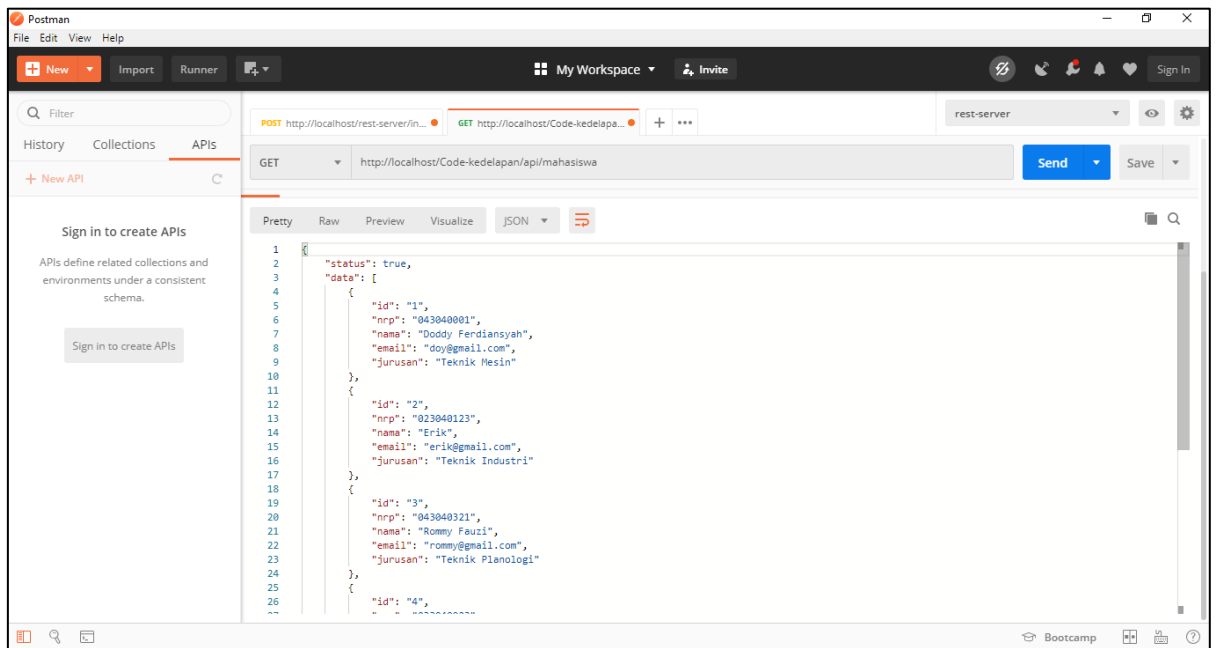
- 7 Selanjutnya buka project <http://localhost/Code-kedelapan/>. Jika berhasil, tampilan akan berubah menjadi seperti ini:

	
8	Tahap instalasi Library Rest Server selesai

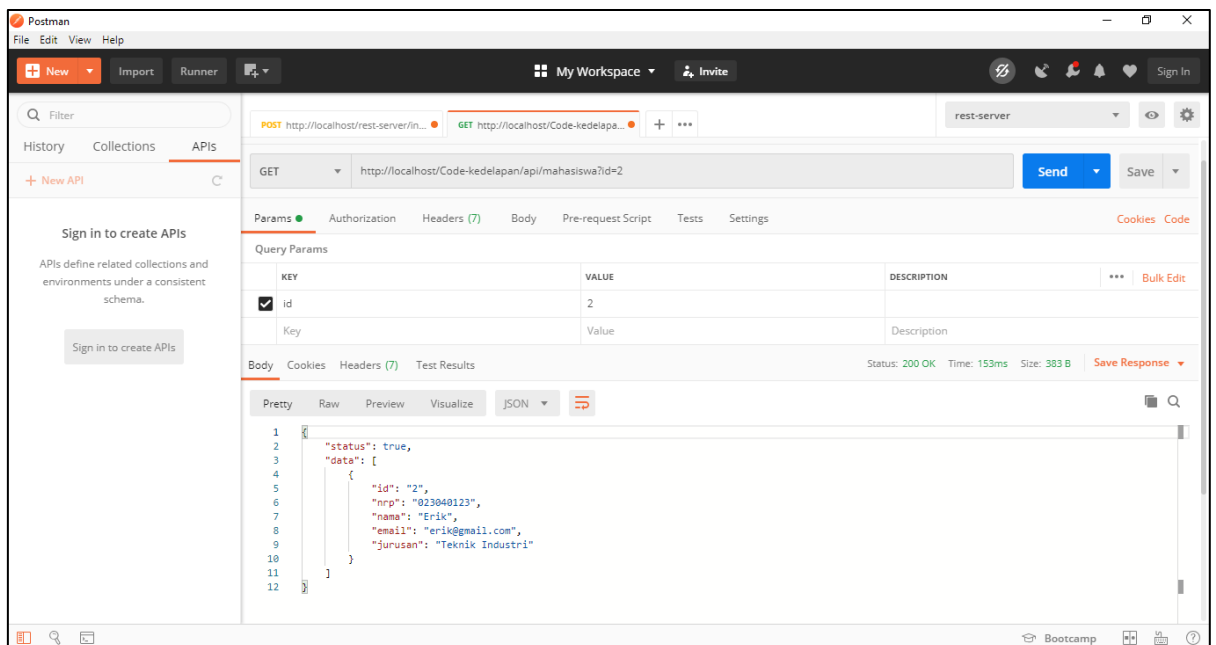
## Praktikum – Bagian 2: Membuat Aplikasi CRUD Restful Codeigniter

Langkah	Keterangan
1	Buatlah file <b>Mahasiswa.php</b> pada <b>controller/api</b> . Selanjutnya tuliskan kode berikut:

	<pre> 1  &lt;?php 2 3  defined('BASEPATH') OR exit('No direct script access allowed'); 4 5  require APPPATH . '/libraries/REST_Controller.php'; 6  require APPPATH . '/libraries/Format.php'; 7  use Restserver\Libraries\REST_Controller; 8 9  class Mahasiswa extends REST_Controller { 10 11     function __construct(\$config = 'rest') { 12         parent::__construct(\$config); 13         \$this-&gt;load-&gt;model('Mahasiswa_model', 'mahasiswa'); 14     } 15 16     function index_get() { 17         \$id = \$this-&gt;get('id'); 18         if (\$id === null) { 19             \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(); 20         } else { 21             \$this-&gt;db-&gt;where('id', \$id); 22             \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(\$id); 23         } 24 25         if (\$mahasiswa) { 26             \$this-&gt;response([ 27                 'status' =&gt; true, 28                 'data' =&gt; \$mahasiswa 29             ], REST_Controller::HTTP_OK); 30         } else { 31             \$this-&gt;response([ 32                 'status' =&gt; true, 33                 'data' =&gt; 'id not found' 34             ], REST_Controller::HTTP_NOT_FOUND); 35         } 36     } </pre>
2	<p>Selanjutnya Buatlah file <b>Mahasiswa_model.php</b> pada model. Selanjutnya tuliskan kode berikut:</p> <pre> 1  &lt;?php 2 3  defined('BASEPATH') OR exit('No direct script access allowed'); 4 5  class Mahasiswa_model extends CI_Model { 6 7     public function getMahasiswa(\$id=null){ 8         if (\$id === null) { 9             return \$this-&gt;db-&gt;get('mahasiswa')-&gt;result_array(); 10         } else { 11             return \$this-&gt;db-&gt;get_where('mahasiswa', ['id' =&gt; \$id])-&gt;result_array(); 12         } 13     } </pre>
3	<p>Buka aplikasi postman, masukkan alamat <a href="http://localhost/Code-kedelapan/api/mahasiswa">http://localhost/Code-kedelapan/api/mahasiswa</a> dengan metode <b>GET</b>. dan jika berhasil maka tampilannya akan seperti ini:</p>



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



4

Untuk fungsi lain **Create**, **Update** dan **Delete**. Lanjutkan kode berikut pada file **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);
        }
    }
}

62 // POST
63 public function index_post(){
64     $data = [
65         'nrp' => $this->post('nrp'),
66         'nama' => $this->post('nama'),
67         'email' => $this->post('email'),
68         'jurusan' => $this->post('jurusan'),
69     ];
70
71     if ($this->mahasiswa->createMahasiswa($data) > 0) {
72         $this->response([
73             'status' => true,
74             'message' => 'new mahasiswa has been created'
75         ], REST_Controller::HTTP_CREATED);
76     } else {
77         $this->response([
78             'status' => false,
79             'message' => 'failed to create new data'
80         ], REST_Controller::HTTP_BAD_REQUEST);
81     }
82 }

```



```

84 // PUT
85 ✓ public function index_put(){
86     $id = $this->put('id');
87 ✓     $data = [
88         'nrp' => $this->put('nrp'),
89         'nama' => $this->put('nama'),
90         'email' => $this->put('email'),
91         'jurusan' => $this->put('jurusan'),
92     ];
93 ✓     if ($this->mahasiswa->updateMahasiswa($data, $id) > 0) {
94 ✓         $this->response([
95             'status' => true,
96             'message' => 'new mahasiswa has been updated'
97         ], REST_Controller::HTTP_OK);
98 ✓     } else {
99 ✓         $this->response([
100             'status' => false,
101             'message' => 'failed to update data'
102         ], REST_Controller::HTTP_BAD_REQUEST);
103     }
104 }
105 }
106 ?>

```

5

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

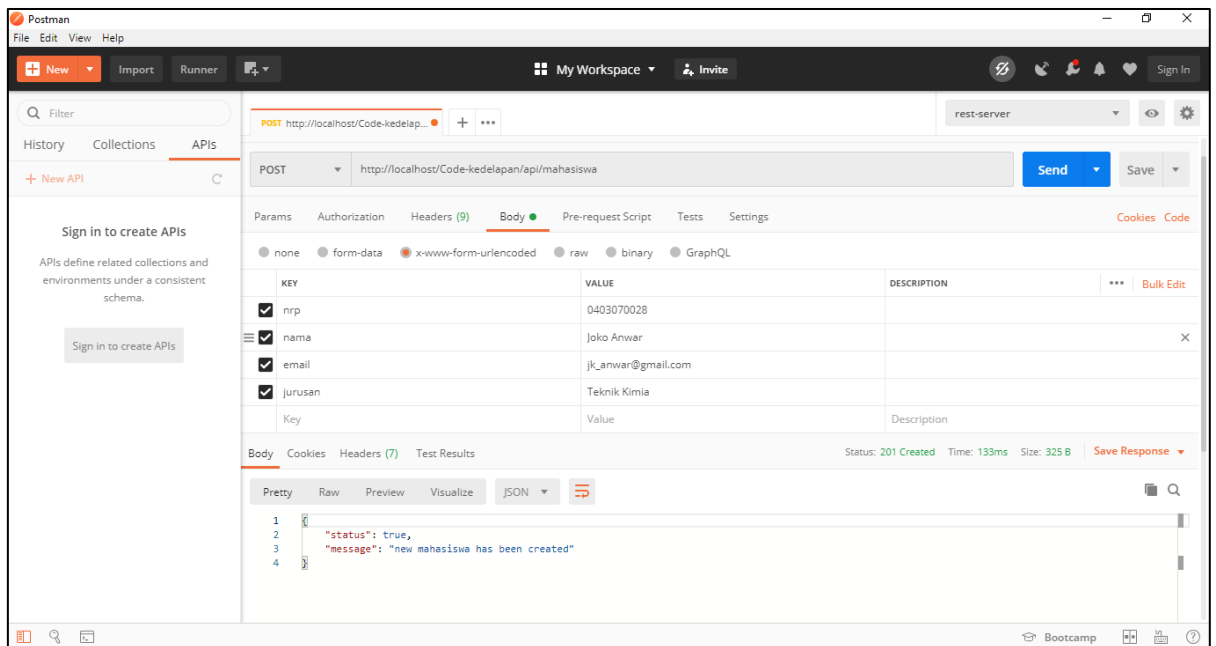
```

14
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 ?>

```

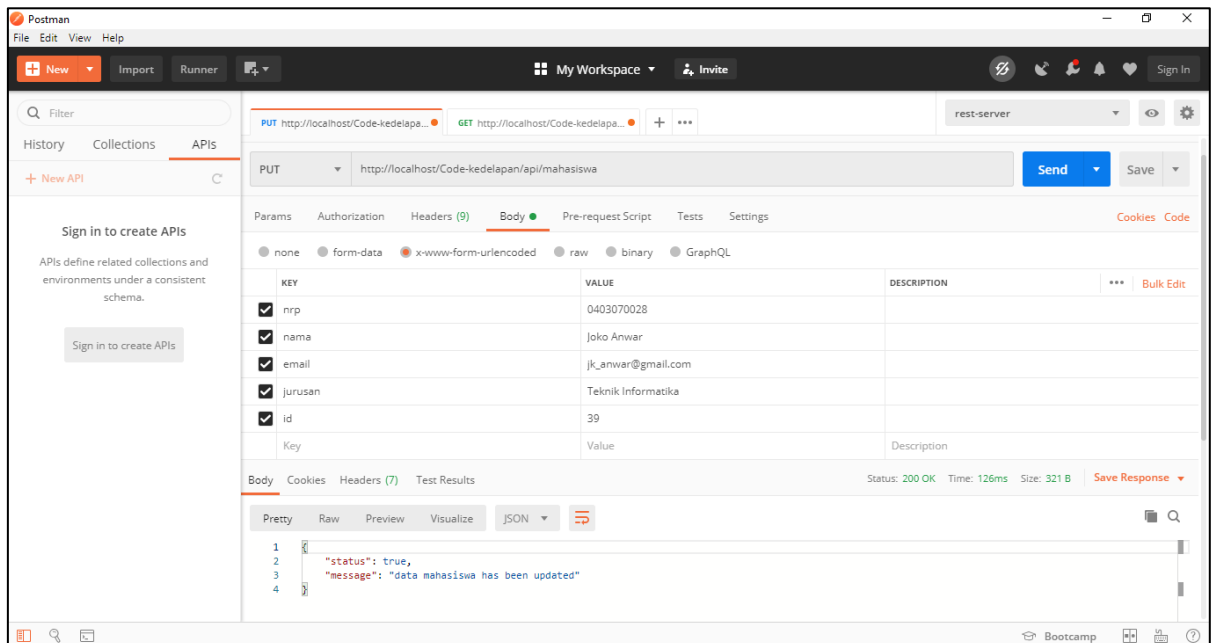
6

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:



7

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



8

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

