

Membuat API dengan Method GET, POST, PUT, DELETE Menggunakan Postman

Disusun untuk memenuhi Tugas Mata Kuliah Teknologi dan Aplikasi Mobile

Dosen Pengampu :

1. Muhaqiqin, S.kom., M.T.I
2. M. Iqbal Parabi, M.T.



Disusun Oleh :

Sinta Nurhalifah	2117051015
Putri Maharani	2117051033
Yola Okta Vina	2157051052
Amelia Agustina Nainggolan	2057051080

JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2022 / 2023

DAFTAR ISI

BAB I PENDAHULUAN	3
1.1 Latar Belakang	3
BAB II PEMBAHASAN	4
2.1 Langkah – Langkah Pembuatan API	4
BAB III PENUTUP	12
3.1 Kesimpulan	12

BAB I PENDAHULUAN

1.1 Latar Belakang

Dengan adanya internet, konsumen dapat dengan mudah mendapatkan informasi mengenai produk dari situs ekspedisi. Konsumen dapat dengan mudah memilih jasa ekspedisi mana yang akan mereka gunakan hanya dengan melihat situs ekspedisi. Konsumen dapat dengan mudah mengecek harga, daerah pengiriman serta tracking pengiriman dalam sebuah ekspedisi. Teknologi Web telah menjadi alat penting untuk mengakses, mengumpulkan, dan mengorganisir data dari Web.

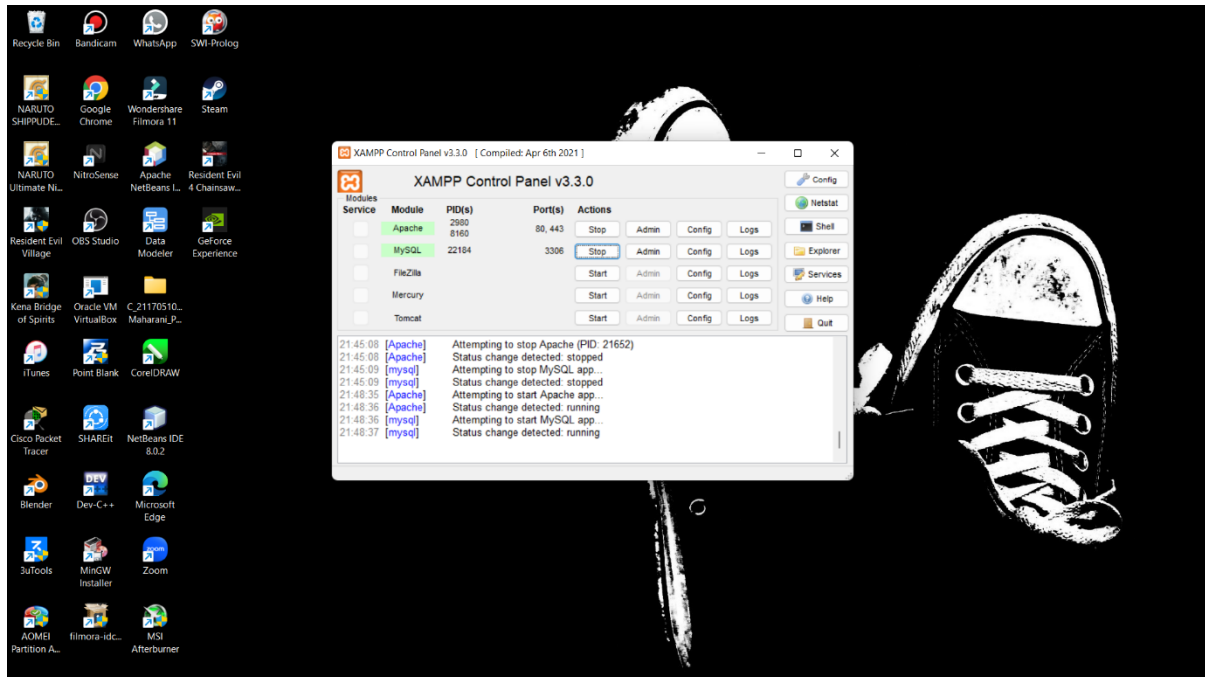
Application Programming Interface (API) adalah sebuah antarmuka yang digunakan oleh pengguna untuk mengakses atau menggunakan sebuah layanan dalam jaringan Internet. Terkadang, ada sejumlah endpoints dalam API yang hanya dikhususkan untuk pengguna dengan tingkat otorisasi tertentu. Contohnya, API endpoint yang memberikan hak akses ke dalam data yang semestinya hanya bisa diakses oleh pengguna dengan tingkat peran tertentu pada sistem dan endpoint untuk melakukan aktivitas tertentu yang tidak bisa diakses oleh sembarang pengguna.

Pada kesempatan ini, kelompok kami akan memfokuskan pada langkah-langkah yang diperlukan dalam pembuatan API. Beberapa persiapan yang perlu dilakukan meliputi penggunaan server lokal seperti XAMPP dan database, penggunaan Postman untuk menguji API dengan metode GET, POST, PUT, DELETE, serta penggunaan Code Editor untuk membuat kode koneksi ke database dan mengambil data dari tabel dalam format JSON.

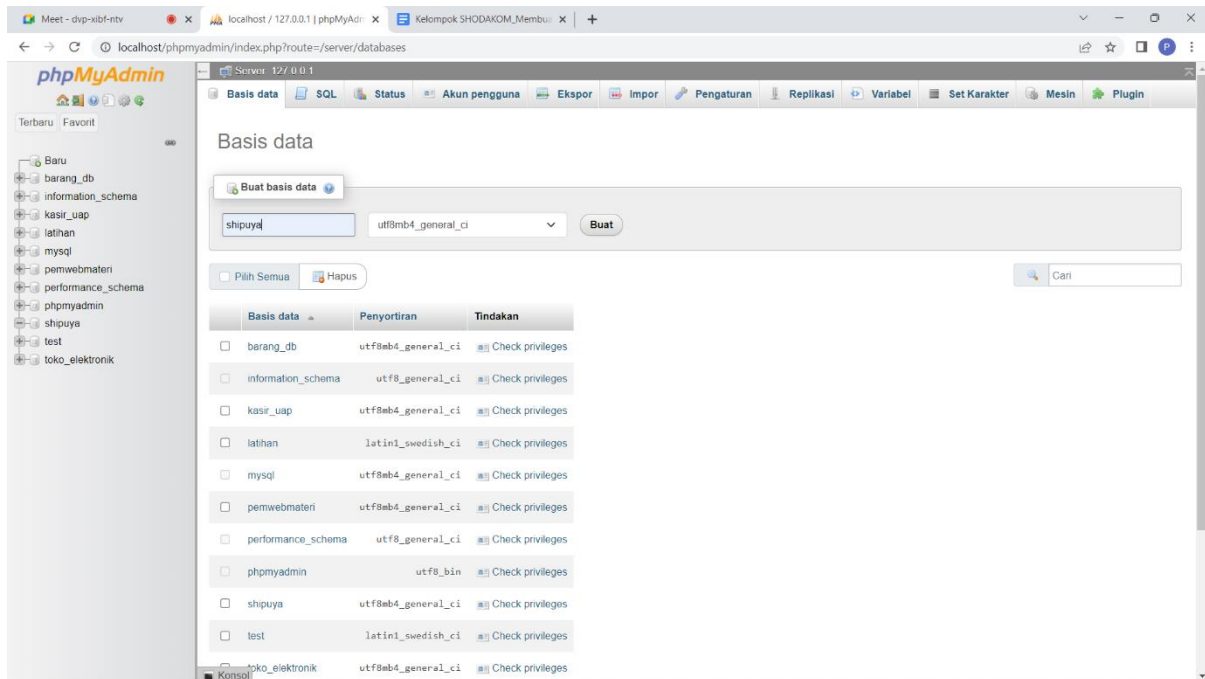
BAB II PEMBAHASAN

2.1 Langkah – Langkah Pembuatan API

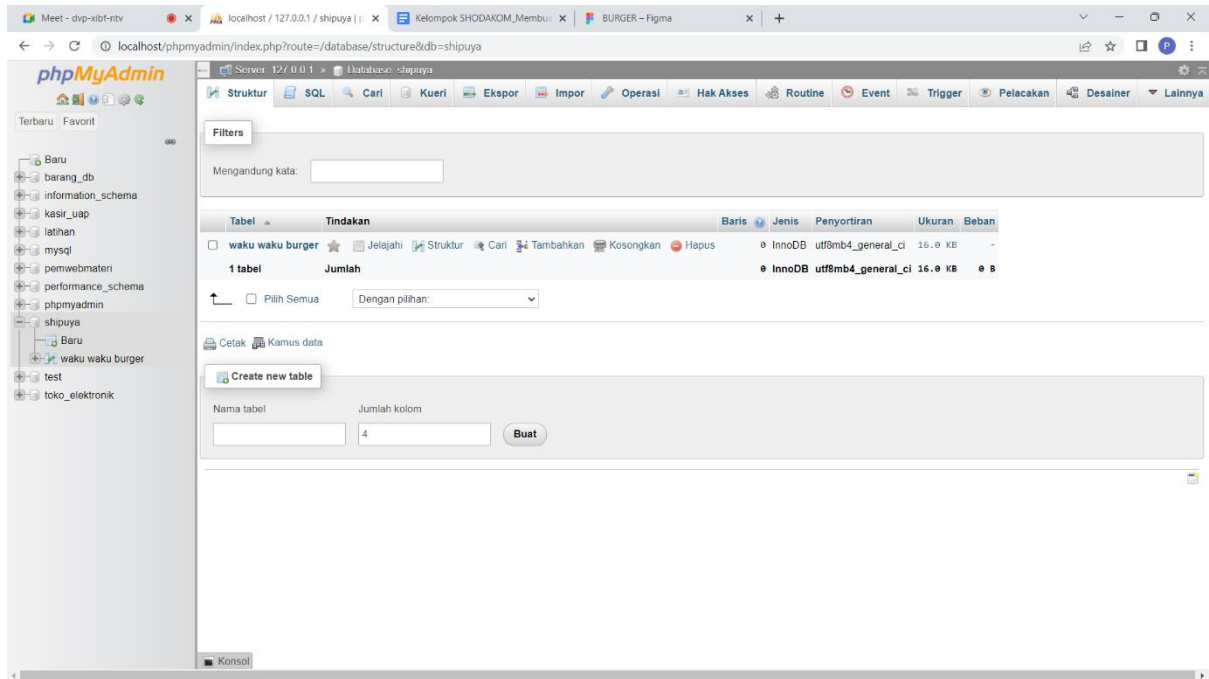
1. Aktifkan XAMPP Control Panel



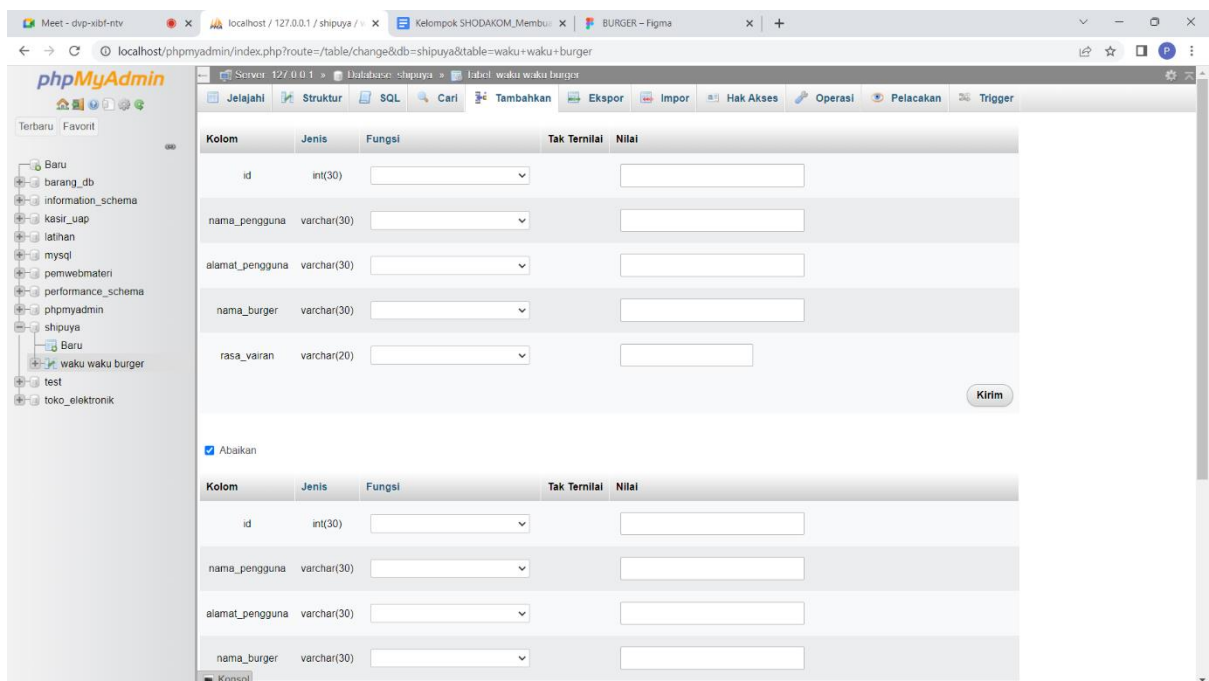
2. Buat database baru pada localhost dan beri nama



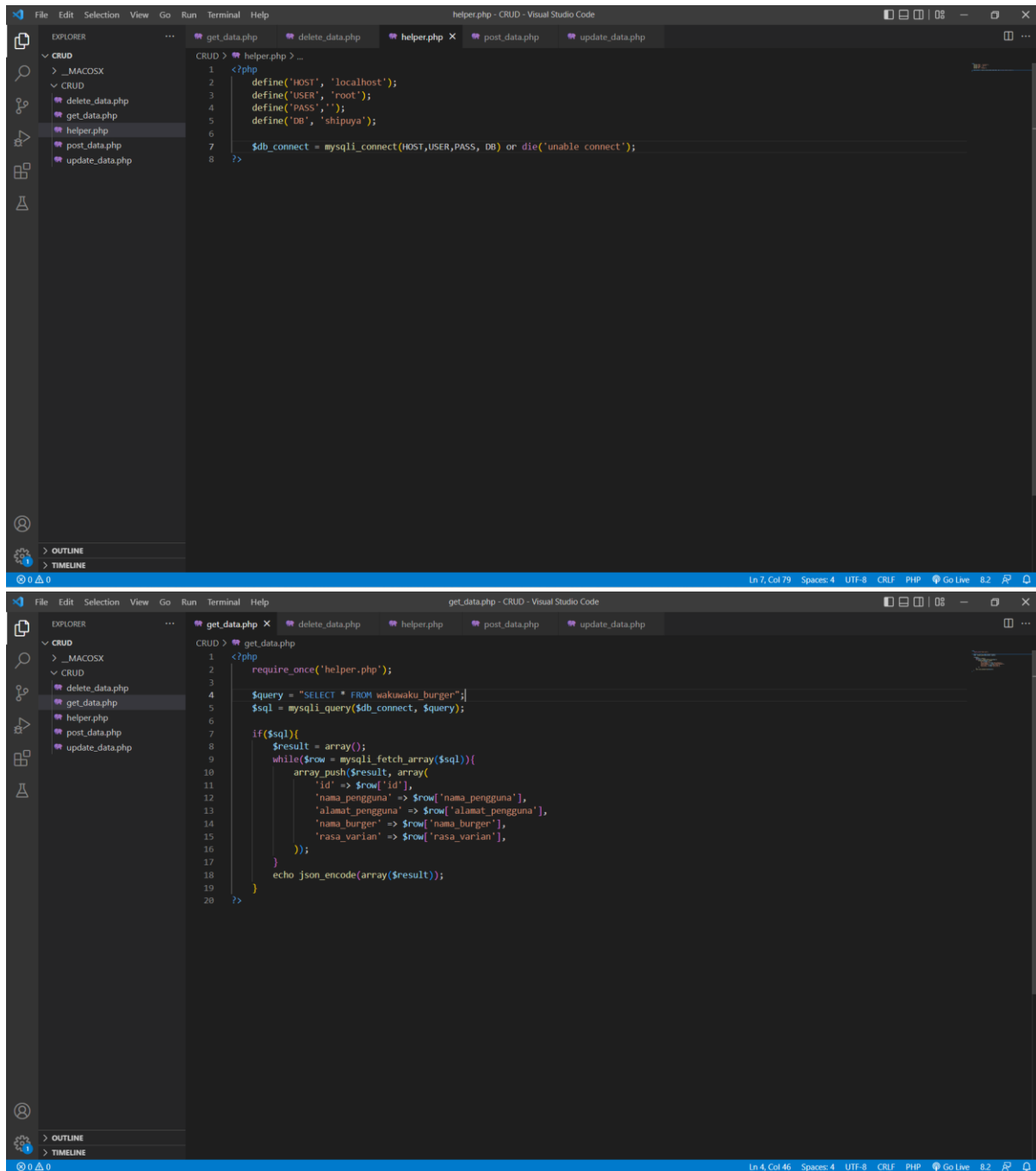
3. Buat table



4. Insert data kedalam tabel yang dibuat



5. Membuat kode program PHP untuk API



```
helper.php - CRUD - Visual Studio Code
1  <?php
2      define('HOST', 'localhost');
3      define('USER', 'root');
4      define('PASS', '');
5      define('DB', 'shipuya');
6
7      $db_connect = mysqli_connect(HOST,USER,PASS, DB) or die('unable connect');
8  ?>
```

```
get_data.php - CRUD - Visual Studio Code
1  <?php
2      require_once('helper.php');
3
4      $query = "SELECT * FROM wakuwaku_burger";
5      $sql = mysqli_query($db_connect, $query);
6
7      if($sql){
8          $result = array();
9          while($row = mysqli_fetch_array($sql)){
10             array_push($result, array(
11                 'id' => $row['id'],
12                 'nama_pengguna' => $row['nama_pengguna'],
13                 'alamat_pengguna' => $row['alamat_pengguna'],
14                 'nama_burger' => $row['nama_burger'],
15                 'rasa_varian' => $row['rasa_varian'],
16             ));
17         }
18         echo json_encode(array($result));
19     }
20  ?>
```

This screenshot shows the Visual Studio Code editor with the 'update_data.php' file open. The Explorer sidebar on the left shows a project structure with a 'CRUD' folder containing 'delete_data.php', 'get_data.php', 'helper.php', 'post_data.php', and 'update_data.php'. The main editor area displays the PHP code for 'update_data.php', which includes a form input, a database update query, and JSON response handling.

```
1 <?php
2
3 require_once('helper.php');
4 parse_str(file_get_contents('php://input'), $value);
5
6 $id = $value['id'];
7 $nama_pengguna = $value['nama_pengguna'];
8 $alamat_pengguna = $value['alamat_pengguna'];
9 $nama_burger = $value['nama_burger'];
10 $rasa_varian = $value['rasa_varian'];
11
12 $query = "UPDATE wakuwaku_burger SET id='$id', nama_pengguna='$nama_pengguna', alamat_pengguna='$alamat_pengguna', nama_burger='$nama_burg
13 $sql = mysqli_query($db_connect, $query);
14
15 if($sql){
16     echo json_encode(array('message' => 'Update!'));
17 } else {
18     echo json_encode(array('message' => 'GAGAL UPDATE!'));
19 }
20 ?>
```

The status bar at the bottom indicates the cursor is at Line 12, Column 37, with 4 spaces, UTF-8 encoding, CRLF line endings, and PHP 8.2.

This screenshot shows the Visual Studio Code editor with the 'post_data.php' file open. The Explorer sidebar on the left shows the same project structure as the first screenshot. The main editor area displays the PHP code for 'post_data.php', which includes a database insert query and JSON response handling.

```
1 <?php
2
3 require_once('helper.php');
4
5 $id = $_POST['id'];
6 $nama_pengguna = $_POST['nama_pengguna'];
7 $alamat_pengguna = $_POST['alamat_pengguna'];
8 $nama_burger = $_POST['nama_burger'];
9 $rasa_varian = $_POST['rasa_varian'];
10
11 $query = "INSERT INTO wakuwaku_burger(id, nama_pengguna, alamat_pengguna, nama_burger, rasa_varian) VALUES ('$id', '$nama_pengguna', '$sala
12 $sql = mysqli_query($db_connect, $query);
13
14 if($sql){
15     echo json_encode(array('message' => 'Created!'));
16 } else {
17     echo json_encode(array('message' => 'GAGAL!'));
18 }
19
20 ?>
```

The status bar at the bottom indicates the cursor is at Line 11, Column 42, with 4 spaces, UTF-8 encoding, CRLF line endings, and PHP 8.2.

BAB III PENUTUP

3.1 Kesimpulan

Maka dapat disimpulkan bahwa ada beberapa langkah persiapan yang perlu dilakukan untuk membuat API. Langkah-langkah tersebut meliputi mengaktifkan XAMPP, membuat database dan membuat tabel dengan data yang relevan. Setelah tahap persiapan selesai, langkah selanjutnya adalah membuat kode PHP untuk menghubungkan API ke database dan mengimplementasikan metode GET, POST, PUT dan DELETE. Pengujian API dilakukan menggunakan Postman, dimana metode yang sesuai digunakan untuk menambahkan, memperbarui atau menghapus data dari tabel shipuya. Dengan adanya API ini akses dan manipulasi data shipuya menjadi lebih mudah.