



VERSI 2.0
24 AGUSTUS, 2023

PIRANTI CERDAS

MODUL 4 : HTTP POST AND GET

TIM PENYUSUN :

ZAMAH SARI, ST., MT.

CLARISSA SANINDITA REIKISYIFA

CHINTYA TRIA DIANA OKTAVIANI

PRESENTED BY: LAB. INFORMATIKA
UNIVERSITAS MUHAMMADIYAH MALANG

PIRANTI CERDAS

PERSIAPAN MATERI

Praktikan mempelajari konsep dasar protokol HTTP post dan get untuk pengiriman data piranti cerdas ke Database Management System (DBMS)

TUJUAN

1. Mahasiswa mampu memahami konsep dasar protokol HTTP post dan get
 2. Mahasiswa mampu membuat program pengiriman data ke database menggunakan protokol HTTP
-

KEBUTUHAN HARDWARE DAN SOFTWARE

1. HARDWARE
 - NodeMCU ESP32
 - Kabel USB
 - Breadboard (Opsional)
 - Kabel Jumper
 - Sensor LDR
 - LED
 - Resistor
 - MPU6050
 2. SOFTWARE
 - Arduino IDE
 - Visual Studio Code: <https://code.visualstudio.com/>
 - XAMPP: <https://www.apachefriends.org/>
-

MATERI POKOK

1. HTTP POST AND GET

HTTP (Hypertext Transfer Protocol) adalah protokol komunikasi yang penting dalam lingkungan web. Dalam pengembangan web, metode HTTP yang umum digunakan adalah POST dan GET. Metode ini memungkinkan interaksi antara peramban web dan server, memfasilitasi pertukaran data dan informasi.

HTTP GET, sebagai salah satu metode dalam HTTP, digunakan untuk mengambil data dari server. Ketika permintaan GET dikirimkan oleh peramban web, server akan merespons dengan data yang diminta. Data ini dapat berupa halaman web, gambar, video, atau berbagai jenis konten

lainnya. Permintaan GET dapat mengandung parameter dalam URL untuk mengarahkan permintaan dengan lebih spesifik, sehingga memungkinkan perolehan data yang diinginkan. Namun, karena parameter URL terbuka, informasi sensitif dapat terekspos jika tidak dikelola dengan benar.

HTTP POST, di sisi lain, digunakan untuk mengirim data dari peramban web ke server. Data ini biasanya tidak ditampilkan langsung dalam URL seperti pada permintaan GET, sehingga lebih cocok digunakan untuk mengirim informasi sensitif seperti kata sandi atau data pribadi. Permintaan POST mengirimkan data dalam badan permintaan HTTP, yang tidak terlihat oleh pengguna biasa. Hal ini membuatnya lebih aman untuk pengiriman data sensitif. Server akan merespons permintaan POST dengan memberikan balasan sesuai, yang dapat berupa konfirmasi sukses atau pesan kesalahan jika ada.

Secara keseluruhan, HTTP GET dan POST adalah dua metode utama dalam protokol HTTP yang memungkinkan komunikasi dan pertukaran data antara peramban web dan server. Pemahaman yang baik tentang perbedaan antara keduanya penting dalam pengembangan web yang efektif dan aman.

2. PENGIRIMAN DATA KE DATABASE DENGAN PROTOKOL HTTP

2.1. DATABASE

Database atau database adalah koleksi terstruktur informasi yang disimpan secara elektronik dalam sistem komputer. Ini memiliki peran penting dalam pengelolaan data untuk berbagai tujuan, dari bisnis hingga penelitian. Komponen utamanya mencakup entitas, atribut, dan relasi di antara entitas tersebut. Database Management System (DBMS) adalah perangkat lunak yang memfasilitasi pengelolaan, akses, dan manipulasi data dalam database. Jenis database seperti relasional, NoSQL, dan berorientasi grafik telah dikembangkan untuk kebutuhan yang bervariasi dalam skala dan kompleksitas.

Dalam piranti cerdas, database penting untuk mengelola data dari perangkat terhubung seperti sensor dan perangkat pintar. Data yang dihasilkan sangat besar dan realtime. Database harus bisa menangani aliran data terus-menerus dan mendukung penyimpanan serta pencarian data efisien. database juga harus bisa memproses data secara paralel dan dapat ditingkatkan seiring bertambahnya perangkat dan data. Penggunaan database yang tepat memungkinkan analisis pola, prediksi, dan pengambilan keputusan yang lebih baik. Namun, perlindungan data dan privasi juga penting dalam pengembangan database untuk piranti cerdas guna menjaga keamanan data dalam

lingkungan yang kompleks dan terhubung.

2.2. PENGIRIMAN DATA KE DATABASE

Pengiriman data ke database melalui protokol HTTP merupakan hal yang umum dalam piranti cerdas. Protokol HTTP digunakan dalam World Wide Web (www) untuk komunikasi data antara server dan klien. Ini memfasilitasi piranti cerdas mengirimkan informasi ke database secara efisien melalui internet, termasuk data sensor, status perangkat, dan interaksi pengguna. Penggunaan HTTP mempermudah akses data dari berbagai platform serta integrasi antara piranti cerdas dan sistem lainnya.

Meski pengiriman data ke database melalui HTTP memiliki keuntungan, seperti implementasi mudah dan interoperabilitas yang baik, kita perlu mempertimbangkan aspek tertentu. Keamanan data saat pengiriman menjadi salah satu pertimbangan penting. Karena data yang dikirim bisa berisi informasi sensitif, perlu ada mekanisme keamanan seperti enkripsi dan otentikasi untuk melindungi data dari akses yang tidak sah. Selain itu, pengiriman data melalui HTTP membutuhkan jaringan yang stabil agar data dapat dikirim dan diterima dengan baik, terutama di lingkungan piranti cerdas yang terus terhubung.

LATIHAN PRAKTIKUM

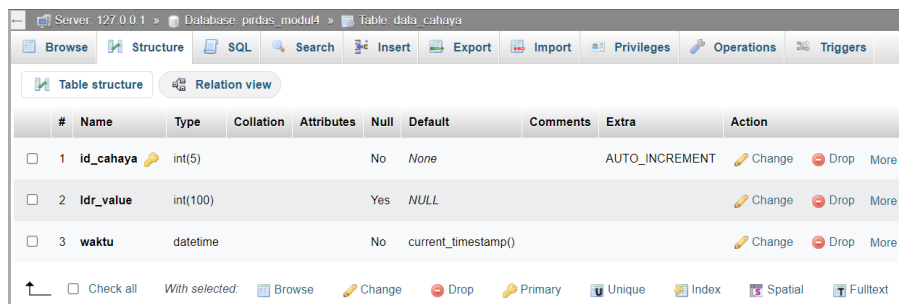
Latihan praktikum wajib dikerjakan setiap orang dengan menggunakan laptop masing-masing. Perangkat fisik yang digunakan dalam latihan dapat bergantian dalam 1 kelompok yang sama.

1. Instalasi VSCODE , XAMPP, dan Pembuatan Database

Tutorial instalasi dapat dilihat pada link sumber berikut:

- Instalasi VSCODE : https://www.youtube.com/watch?v=JPZsB_6yHVo
- Instalasi XAMPP : <https://www.youtube.com/watch?v=VCHXCusltqI>
- Pembuatan Database : <https://www.youtube.com/watch?v=co-xyHRdHRg>

Berikut adalah database yang digunakan untuk kegiatan praktikum ini:

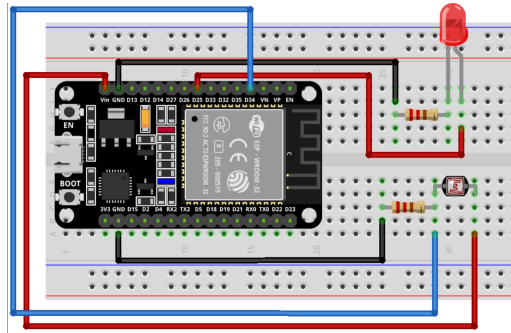


The screenshot shows a database management interface with a table structure view. The table is named 'data_cahaya' and has three columns: 'id_cahaya', 'ldr_value', and 'waktu'. The 'id_cahaya' column is an integer with a length of 5, is not null, and has a default value of 'None'. The 'ldr_value' column is an integer with a length of 100, is nullable, and has a default value of 'NULL'. The 'waktu' column is a datetime type, is not null, and has a default value of 'current_timestamp()'. The interface also includes tabs for 'Table structure', 'Relation view', 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', and 'Triggers'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_cahaya	int(5)			No	None		AUTO_INCREMENT	Change Drop More
2	ldr_value	int(100)			Yes	NULL			Change Drop More
3	waktu	datetime			No	current_timestamp()			Change Drop More

2. Pembuatan Rangkaian Sensor Cahaya

Berikut adalah ilustrasi rangkaian sensor cahaya



3. Pembuatan Server Scripting Menggunakan PHP

Berikut adalah kode PHP untuk pengiriman data

```

1  <?php
2
3      $servername = "localhost";
4      $username = "root";
5      $password = "";
6      $dbname = "pirdas_modul4";
7
8
9      if( $_GET["ldr_value"])
10         $ldr_value = $_GET["ldr_value"];
11
12     $conn = new mysqli($servername, $username, $password, $dbname);
13
14     if ($conn->connect_error)
15     {
16         die("Connection failed: " . $conn->connect_error);
17     }
18
19     $sql = "INSERT INTO data_cahaya (ldr_value) VALUES (". $ldr_value . ")";
20
21
22     if ($conn->query($sql)===TRUE) {
23         echo "DATA LDR VALUE BERHASIL DITAMBAH";
24     }
25     else{
26         echo "Error: " . $sql . "<br>" . $conn->error;
27     }
28
29     $conn->close();
30     exit();
31
32  ?>

```

Setelah itu lakukan testing untuk memeriksa apakah data ldr_value berhasil tertambah pada database yaitu dengan paste URL ini pada browser:

http://localhost/pirdas_modul4/cahaya.php?ldr_value=40

Keterangan:

- localhost → merupakan nama server, atau bisa juga diganti dengan IP server
- pirdas_modul4/ → merupakan nama directory / path tempat menaruh script PHP untuk menambah data cahaya
- cahaya.php → merupakan script PHP yang dijalankan atau direquest oleh client
- ldr_value=400 → merupakan parameter yang digunakan untuk menyimpan nilai yang dilewatkan ke server. Jika terdapat lebih dari satu parameter, maka antara parameter satu dengan yang lain dipisah tanda "&"

Kemudian refresh database phpmyadmin dan periksa apakah data bertambah.

4. Kode Program ESP32

- a. Berikut adalah kode program ESP32 pada Arduino IDE untuk mencoba rangkaian yang sudah dibuat

```
int LDR_Val = 0;
int SENSOR = 34;
int LED = 25;
void setup() {
    Serial.begin(9600);
    pinMode(LED, OUTPUT);
}
void loop() {
    LDR_Val = analogRead(SENSOR);
    Serial.print("LDR Output: ");
    Serial.println(LDR_Val);
    if(LDR_Val > 100) {
        Serial.println(" Cahaya Terang ");
        digitalWrite(LED, LOW);
    }
    else {
        Serial.println("Cahaya Gelap ");
        digitalWrite(LED, HIGH);
    }
    delay(1000);
}
```

- b. Berikut adalah kode program ESP32 untuk pengiriman data ke database yang sudah dibuat

```
1  #include <Arduino.h>
2  #include <HTTPClient.h>
3  #include <WiFi.h>
4
5  #define SENSORPIN 34
6  #define LEDPIN 25
7
8  const char* ssid = "TULIS NAMA SSD HOTSPOT KALIAN";
9  const char* password = "TULIS PASSWORD DARI HOTSPOT KALIAN";
10 const char* serverName = "http://172.20.10.6/pirdas_modul4/cahaya.php";
11
12 void setup() {
13     Serial.begin(9600);
14     pinMode(LEDPIN, OUTPUT);
15     delay(1000);
16
17     // Connect to Wi-Fi network
18     WiFi.begin(ssid, password);
19     while (WiFi.status() != WL_CONNECTED) {
20         delay(1000);
21         Serial.println("Connecting to WiFi...");
22     }
23
24     String wifiName = WiFi.SSID();
25     String localWifiIP = WiFi.localIP().toString();
26     Serial.println();
27     Serial.println("Connected to " + wifiName);
28     Serial.println("Local IP WIFI: " + localWifiIP);
29 }
30
31 void loop() {
32     int ldr_value = analogRead(SENSORPIN);
```

```

34  if(ldr_value > 100) {
35      Serial.println(" Cahaya Terang ");
36      digitalWrite(LEDPIN, LOW);
37      Serial.print("Intensitas Cahaya: ");
38      Serial.print(ldr_value);
39      Serial.println();
40  }
41  else {
42      Serial.println("Cahaya Gelap ");
43      digitalWrite(LEDPIN, HIGH);
44      Serial.print("Intensitas Cahaya: ");
45      Serial.print(ldr_value);
46      Serial.println();
47  }
48  delay(1000);
49  String url = serverName;
50  url += "?ldr_value=";
51  url += String(ldr_value);
52
53  HTTPClient http;
54  http.begin(url);
55  int httpResponseCode = http.GET();
56  if (httpResponseCode == 200) {
57      Serial.println("Data sent successfully!");
58  } else {
59      Serial.print("Error code: ");
60      Serial.println(httpResponseCode);
61  }
62  http.end();
63  delay(10000);
64  }

```

NB. Pastikan ESP32 dan laptop terkoneksi pada jaringan WiFi yang sama dan pastikan penyimpanan file berada dalam folder xampp/htdocs

LEMBAR KERJA PRAKTIKUM

Kegiatan 1:

1. Buatlah rangkaian untuk sensor gerak yang mendeteksi kemiringan dan sensor cahaya untuk mendeteksi intensitas cahaya dalam ruangan. Kemudian tampilkan hasil pembacaan data sensor melalui serial monitor Arduino IDE yang terhubung ke NodeMCU ESP32
2. Komponen yang diperlukan antara lain:
 - a. NodeMCU ESP32
 - b. Sensor LDR
 - c. Sensor MPU 6050
 - d. Breadboard, kabel jumper
 - e. Resistor 10k Ohm.
3. Buatlah program untuk menjalankan rangkaian ini **yang tidak terkoneksi dengan database**

Ketentuan program antara lain:

- Jika hasil pembacaan sensor LDR berada pada rentang 0 sampai 50, tampilkan tulisan “keadaan gelap” pada serial monitor. Jika berada pada rentang 50 sampai 500 “keadaan redup” dan jika

berada pada rentang 500 hingga 2000 kondisi “cahaya terang” dan jika lebih dari 2000 kondisi “sangat terang”.

- Jika sensor MPU6050
 - a. Sumbu x > 10 sensor miring “kiri”
 - b. Sumbu x < -10 sensor miring “kanan”
 - c. Sumbu y > 10 sensor miring “depan”
 - d. Sumbu y < -10 sensor miring “belakang”

Kegiatan 2:

1. Buatlah server scripting menggunakan bahasa pemrograman PHP untuk studi kasus dengan sensor **MPU6050**
2. Kirimkan data yang ditangkap ke DBMS
3. Minimal kolom yang harus ada dalam tabel pada database adalah:
 - id_sensor_gerak mandatory, auto increment
 - angka_sumbu
 - kemiringan (hanya boleh kanan, kiri, depan, atau belakang saja)

Notes:

1. Akan terdapat nilai tambah jika praktikan mampu membuat aplikasi berbasis website yang dapat mencatat data sensor
2. Akan terdapat nilai tambah jika praktikan menambahkan kolom yang relevan dengan program yang dibuat.

PERINGATAN: Terdapat Kemiripan Source Code Yang Signifikan Dengan Praktikan Lain, Nilai Maksimal D

RUBRIK PENILAIAN PRAKTIKUM

Detail	Bobot Nilai
Pemahaman Materi	40%
Ketepatan Jawaban	30%
Program dapat berjalan tanpa error	15%
Tugas Pekan Materi	15%