

### 1. Diagram Arsitektur IoT untuk Aplikasi Smart Home

Diagram ini mengilustrasikan arsitektur sistem IoT sederhana untuk aplikasi Smart Home. Sistem ini terdiri dari beberapa komponen utama: Perangkat IoT, Jaringan Lokal (Gateway), Platform Cloud IoT, dan Aplikasi Pengguna (Smartphone).

Penjelasan Alur Kerja:

#### 1. Perangkat IoT (Things):

- Sensor Suhu: Mengirimkan data suhu secara terus-menerus ke gateway.
- Lampu Pintar: Terhubung ke gateway melalui Zigbee/WiFi. Lampu ini dapat menerima perintah ON/OFF.
- Kamera Keamanan: Mengirimkan video stream ke gateway dan dapat menerima perintah seperti "Putar" atau "Rekam".

#### 2. Jaringan Lokal (Gateway / Hub):

- Berfungsi sebagai jembatan antara perangkat IoT dengan internet.
- Menerima data dari semua sensor dan perangkat, lalu meneruskannya ke platform cloud.
- Menerima "Kirim Perintah" dari cloud untuk diteruskan ke perangkat IoT (misalnya, mematikan lampu).

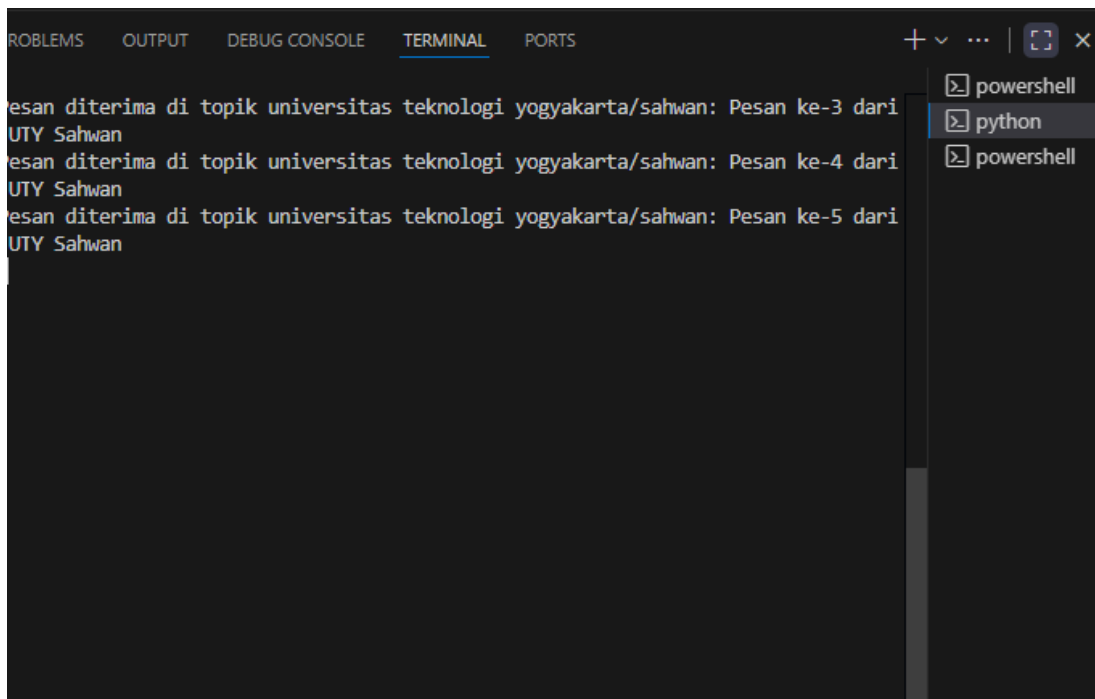
#### 3. Platform Cloud IoT:

- Menerima dan menyimpan semua data yang dikirim oleh gateway. ○ Melakukan sinkronisasi data dengan aplikasi smartphone pengguna.
- Mengirimkan perintah dari pengguna (misalnya, "Kontrol Lampu") ke gateway.

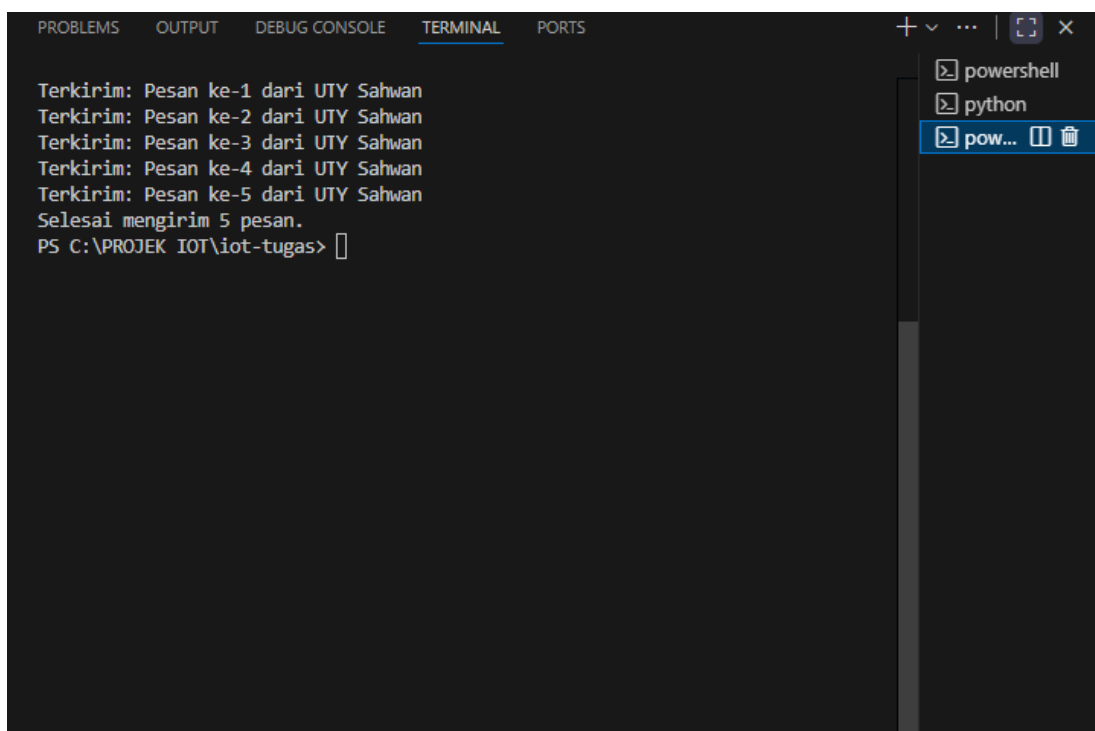
#### 4. Aplikasi Smartphone (Pengguna):

- Menjadi antarmuka bagi pengguna untuk memonitor (melihat data suhu) dan mengontrol (menyalakan/mematikan lampu) perangkat IoT dari jarak jauh.

## 2. Dokumentasi Publish & Subscribe dengan MQTT



The screenshot shows a VS Code terminal window with the 'TERMINAL' tab selected. The terminal displays three lines of received MQTT messages: 'Pesan diterima di topik universitas teknologi yogyakarta/sahwan: Pesan ke-3 dari UTY Sahwan', 'Pesan diterima di topik universitas teknologi yogyakarta/sahwan: Pesan ke-4 dari UTY Sahwan', and 'Pesan diterima di topik universitas teknologi yogyakarta/sahwan: Pesan ke-5 dari UTY Sahwan'. On the right side, a context menu is open, showing options to open the terminal in 'powershell', 'python', or 'powershell'.



The screenshot shows a VS Code terminal window with the 'TERMINAL' tab selected. The terminal displays five lines of sent MQTT messages: 'Ter kirim: Pesan ke-1 dari UTY Sahwan', 'Ter kirim: Pesan ke-2 dari UTY Sahwan', 'Ter kirim: Pesan ke-3 dari UTY Sahwan', 'Ter kirim: Pesan ke-4 dari UTY Sahwan', and 'Ter kirim: Pesan ke-5 dari UTY Sahwan'. The sixth line says 'Selesai mengirim 5 pesan.' followed by a PowerShell prompt 'PS C:\PROJEK IOT\iot-tugas>'. On the right side, a context menu is open, showing options to open the terminal in 'powershell', 'python', or 'pow...' (PowerShell).

Implementasi ini menggunakan protokol MQTT untuk komunikasi antara publisher (pengirim pesan) dan subscriber (penerima pesan). Topik MQTT yang digunakan sesuai dengan tugas, yaitu: "Universitas Teknologi Yogyakarta/sahwan".

### Sesi Publisher (Pengiriman Pesan)

Skrip `app_original_topic.py` berhasil dijalankan untuk mengirim 5 pesan berurutan ke broker MQTT. Setiap pesan dikirim dalam format JSON dan berisi informasi seperti nomor pesan, isi, timestamp, nama pengirim, universitas, dan NIM.

### Sesi Subscriber (Penerimaan Pesan)

Skrip subscriber\_original\_topic.py dijalankan untuk berlangganan (subscribe) ke topik yang sama. Skrip ini berhasil menerima kelima pesan yang dikirim oleh publisher secara real-time, membuktikan bahwa komunikasi berjalan dengan sukses.

### 3. Penerapan Konsep IoT pada Aplikasi Smart Classroom

Konsep arsitektur IoT dan protokol publish-subscribe (MQTT) yang telah diimplementasikan sangat relevan untuk dikembangkan menjadi aplikasi Smart Classroom. Sistem ini bertujuan untuk mengotomatisasi fasilitas kelas, meningkatkan efisiensi energi, dan menciptakan lingkungan belajar yang lebih interaktif.

Contoh Skenario Implementasi:

- Presensi Otomatis:
  - Publisher: Sebuah card reader RFID/NFC di pintu kelas mempublikasikan NIM mahasiswa ke topik kampus/kelas\_A101/presensi saat kartu ditempelkan.
  - Subscriber: Sistem informasi akademik universitas berlangganan ke topik tersebut untuk mencatat kehadiran secara otomatis.
- Kontrol Lingkungan Kelas Cerdas:
  - Publisher: Sensor suhu dan cahaya secara periodik mempublikasikan data ke topik kampus/kelas\_A101/lingkungan.
  - Subscriber: Sistem manajemen gedung menerima data ini. Jika suhu ruangan melebihi 25°C, sistem secara otomatis mengirim perintah ke unit AC. Jika tidak ada gerakan terdeteksi selama 15 menit, sistem akan mematikan lampu dan AC.
- Mode Presentasi Pintar:
  - Publisher: Dosen menekan tombol "Mulai Presentasi" pada aplikasi di smartphone. Perintah ini dipublikasikan ke topik kampus/kelas\_A101/mode.
  - Subscriber: Lampu, proyektor, dan tirai jendela yang berlangganan topik ini akan merespons secara serentak: lampu akan meredup, proyektor menyala, dan tirai otomatis tertutup.

Dengan demikian, arsitektur yang sama dapat diadaptasi untuk menciptakan ekosistem perangkat yang saling terhubung di dalam ruang kelas, menjadikan proses belajar mengajar lebih efisien dan modern.