**Pengembangan Gateway Untuk Kendali Perangkat Internet of Things Berbasis Advance Message Queueing Protocol**

**Mikrokontroller**  
Arduino, Raspberry, dll

**IoT Gateway**

Sensor

**Cloud Platform**

1. **Cloud Platform**

* Mengkonfigurasi IoT Gateway yang terhubung meliputi:
  + Id IoT Gateway
  + Lokasi (Lat dan Lang)
  + Jumlah mikon yang terhubung ke IoT Gateway lengkap dengan sensor yang terpasang
* Menyimpan data-data yang dikirimkan IoT Gateway
* Memberikan informasi perintah ke IoT Gateway, yaitu tertuju kepada mikon yang terhubung. Sebagai contoh: menyalakan lampu

1. **IoT Gateway**

IoT Gateway memiliki platform berbasis web yang memiliki fungsi sebagai berikut:

* Memanajemen mikrokontroller yang akan terhubung didaftarkan ke sistem pada IoT Gateway. Attribut yang didaftarkan antara lain:
  + Id Mikrokontroller
  + Sensor yang digunakan : pengontrol lampu, water meter, weather station dan sensor parkir
  + Token
* Monitoring mikrokontroller melalui data log meliputi tanggal, id mikrokontroller dan data sensor
* Konfigurasi koneksi IoT Gateway ke internet melalui LAN/Modem.
* Konfigurasi ethernet 0 dan ethernet 1
* Konfigurasi wifi
* Terdapat fitur untuk mengirimkan perintah ke mikrokontroller. Perintah bisa dari Cloud Platform (Proses D) atau langsung dari platform IoT Gateway.
* Perintah tersebut akan tersimpan di datalog perintah.
* Secara berkala mikrokontroller akan merequest informasi perintah dari IoT Gateway (Proses B)

1. **Proses A**

* Mikrokontroller secara berkala akan mengirimkan data sensor ke sistem IoT Gateway
* Pengiriman data melalui kabel UTP atau Wifi
* Pengiriman data menggunakan protokol AQMP (Advance Message Queueing Protocol) dengan mengakses IP Address IoT Gateway
* Data yang dikirim oleh mikrokontroller akan tersimpan di datalog IoT Gateway

1. **Proses B**

* Mikrokontroller secara berkala akan merequest informasi perintah dari IoT Gateway yaitu :
  + Mikrokontroller mendapatkan update data perintah dari IoT Gateway berdasarkan sensor yang terpasang
* Mikrokontroller akan mengeksekusi perintah sesuai dengan informasi yang didapat
  + Sensor pengontrol lampu: jika perubahan status lampu dari 0 menjadi 1 maka, mikrokontroller akan menghidupkan lampu. Begitu juga sebaliknya.
  + Sensor water meter: mikon akan mengirimkan data volume air yang sedang terbaca
  + Sensor weather station: mikon akan mengirimkan data suhu, kelembapan dan intensitas cahaya yang sedang terbaca
  + Sensor parkir: mikon akan mengirimkan status 1 yaitu jika ada kendaraan dan 0 jika tidak ada kendaraan yang parkir

1. **Proses C**

* IoT Gateway secara berkala akan mengirimkan datalog ke Cloud Platform.
* Data yang dikirimkan meliputi mikrokontroller yang terhubung, data sensor dari setiap mikrokontroller.

1. **Proses D**

* IoT Gateway secara berkala merequest informasi perintah dari Cloud Platform.
* Informasi perintah dari Cloud Platform akan disimpan kedalam datalog perintah.

**Worksheet Pengerjaan Riset**

1. Mikrokontroller 🡨🡪 Sensor

|  |  |  |  |
| --- | --- | --- | --- |
| **Target** | **Aktifitas** | **Kebutuhan** | **Validasi** |
| Pembuatan alat pengontrol lampu | * Mencari alat-alat yang dibutuhkan * Merakit alat-alat * Mencoba menyalakan lampu dan mematikan lampu * Membaca status lampu nyala atau mati | * Arduino UNO * Raspberry Pi * Relay * Lampu * Kabel * Breadboard | * Node MCU versi Arduino sudah jadi * Node MCU versi Raspberry belum jadi * Casing belum jadi |
| Pembuatan alat water meter | * Mencari alat-alat yang dibutuhkan * Merakit alat-alat * Mengecek pembacaan volume air * Menampilan volume air ke LCD * Membuat casing alat | * Arduino UNO * Raspberry Pi * Flow Meter * Kabel * Breadboard * GSM Modem * LCD 16x2 * Pipa * Wadah air * Pompa | * Node MCU versi Arduino sudah jadi * Node MCU versi Raspberry belum jadi * Casing belum jadi |
| Pembuatan alat weather station | * Mencari alat-alat yang dibutuhkan * Merakit alat-alat * Mengecek pembacaan suhu, pH dan intensitas cahaya * Menampilkan suhu, pH dan intensitas cahaya ke LCD TFT * Membuat casing alat | * Arduino UNO * Raspberry Pi * Kabel * Breadboard * Sensor Suhu * Sensor pH * Sensor Intensitas Cahaya * LCD TFT | * Node MCU versi Arduino belum jadi * Node MCU versi Raspberry belum jadi * Casing belum jadi |
| Pembuatan alat sensor parkir | * Mencari alat-alat yang dibutuhkan * Merakit alat-alat * Mengecek pembacaan objek yang melewati sensor ultrasonic * Membuat casing alat | * Arduino UNO * Raspberry Pi * Kabel * Breadboard * Sensor Suhu * Sensor Ultrasonic * LED | * Node MCU versi Arduino belum jadi * Node MCU versi Raspberry belum jadi * Casing belum jadi |

1. Mikrokontroller 🡪 IoT Gateway

|  |  |  |  |
| --- | --- | --- | --- |
| **Target** | **Aktifitas** | **Kebutuhan** | **Validasi** |
| Konfigurasi ethernet card di Arduino UNO | * Mencari alat-alat yang dibutuhkan * Merakit alat-alat * Mengecek koneksi ethernet dari arduino ke komputer/pc/laptop * Membuat sample API/AQMP * Mengkases API/AQMP yang disediakan komputer/pc/laptop | * Arduino UNO * Kabel * Breadboard * Ethernet shield * Laptop/pc | * Node MCU versi Arduino belum jadi |
| Konfigurasi ethernet card dari raspberry | * Mencari alat-alat yang dibutuhkan * Menambahkan ethernet port * Mengecek koneksi ethernet 1 dan 2 dari raspberry ke komputer/pc/laptop * Membuat sample API/AQMP * Mengkases API/AQMP yang disediakan komputer/pc/laptop | * Raspberry Pi * Ethernet module * Kabel * Breadboard * Ethernet shield * Komputer | * Node MCU versi Raspberry belum jadi |
| Pengiriman data sensor dari arduino/raspberry ke komputer | * Membuat sample API/AQMP untuk menyimpan data sensor lampu (hidup/mati), water meter (volume), weather (suhu, intensitas cahaya, kelembapan), parkir (jarak) * Mencoba pengiriman data sensor lampu ke komputer melalui API/AQMP * Mencoba pengiriman data sensor flow meter ke komputer melalui API/AQMP * Mencoba pengiriman data weather station (suhu, intensitas cahaya, kelembapan) ke komputer melalui API/AQMP * Mencoba pengiriman data sensor parkir (ultrasonic) ke komputer melalui API/AQMP   Sensor  Node MCU  IoT Gateway | * Node MCU sensor lampu * Node MCU sensor water meter * Node MCU sensor weather station * Node MCU sensor parkir * Kabel UTP * Switch * Komputer | Terkoneksi Node MCU ke komputer |

1. IoT Gateway 🡪 Mikrokontroller

|  |  |  |  |
| --- | --- | --- | --- |
| **Target** | **Aktifitas** | **Kebutuhan** | **Validasi** |
| Pembuatan *interface* permintaan data dari IoT Gateway ke Node MCU | * Membuat *user interface* untuk permintaan data ke Node MCU. * Setiap *action* dari *user interface* akan disimpan kedalam *datalog command*   user    user interface  datalog command | * IoT Gateway * Komputer * Kabel UTP | * User Interface dapat digunakan dan dapat menyimpan setiap *action* kedalam *datalog command* |
| Pembuatan API/AQMP untuk menangani permintaan *datalog command* dari Node MCU | API/AQMP *datalog command* yang diakses dari Node MCU:   * informasi kondisi lampu (hidup/mati) * informasi permintaan data volume * informasi permintaan data suhu, intensitas cahaya dan kelembapan * informasi permintaan data jarak | * Komputer | * API/AQMP terbentuk |
| Rikues API/AQMP secara rutin dari Node MCU ke komputer | * Setiap 5 detik sekali Node MCU akan merikues *datalog command* ke IoT Gateway melalui API/AQMP, yang sesuai dengan sensor yang terpasang. * Kemudian Node MCU akan mengecek data sensor, selanjutnya dikirim ke IoT Gateway. * Data yang diterima oleh IoT Gateway akan disimpan kedalam IoT Gateway *datalog info*   Node MCU  IoT Gateway | * Komputer * Node MCU * Kabel UTP | * Node MCU dapat mengecek datalog command dan dapat mengirimkan kembali data sensor ke IoT Gateway |

1. IoT Gateway 🡪 Cloud Platform

|  |  |  |  |
| --- | --- | --- | --- |
| **Target** | **Aktifitas** | **Kebutuhan** | **Validasi** |
| Pembuatan API/REST di Cloud Platform | API/REST untuk menerima data-data dari IoT Gateway yang terdiri dari:   * Identitas IoT Gateway * Node MCU yang terpasang pada IoT Gateway * Sensor yang terpasang pada setiap Node MCU   IoT Gateway  Node MCU  Sensor | * Hosting * IoT Gateway * Komputer | * Terbentuknya API/REST didalam Cloud Platform |
| Pengiriman data dari IoT Gateway ke Cloud Platform | * IoT Gateway secara berkala (10 detik) akan mengirimkan *datalog info* yang berisi pencatatan sensor dari Node MCU ke Cloud Platform dengan mengakses API/REST yang telah dibuat | * IoT Gateway * Hosting * Node MCU * Sensor * Komputer * Kabel UTP * Koneksi Internet | * IoT Gateway berhasil mengirimkan data ke Cloud Platform |

1. Cloud Platform 🡪 IoT Gateway

|  |  |  |  |
| --- | --- | --- | --- |
| **Target** | **Aktifitas** | **Kebutuhan** | **Validasi** |
| Pembuatan interface untuk mengakses data ke IoT Gateway | * Membuat *user interface* untuk permintaan data ke IoT Gateway. * Setiap aksi dari *user interface* akan disimpan kedalam *datalog command* | * Hosting * Komputer * Koneksi Internet | * User Interface dapat digunakan dan dapat menyimpan setiap *action* kedalam *datalog command* |
| Pembuatan API/REST *datalog command* yang akan diakses oleh IoT Gateway | API/REST *datalog command* yang diakses dari IoT Gateway:   * informasi Node MCU terpasang dengan kondisi lampu (hidup/mati) * informasi Node MCU terpasang dengan permintaan data volume * informasi Node MCU terpasang dengan permintaan data suhu, intensitas cahaya dan kelembapan * informasi Node MCU terpasang dengan permintaan data jarak | * Hosting * IoT Gateway * Komputer * Koneksi Internet | * API/REST terbentuk |
| Rikues API/REST secara rutin dari IoT Gateway ke Cloud Platform | * Setiap 5 detik sekali IoT Gateway akan merikues *datalog command* yang disediakan Cloud Platform melalui API/REST, yang sesuai dengan Node MCU dan sensor yang terpasang. * Kemudian IoT Gateway akan menyimpan data yang diterima (dari Cloud Platform) kedalam IoT Gateway *datalog command*   Cloud Platform  IoT Gateway  datalog command | * Hosting * Komputer * IoT Gateway * Node MCU * Kabel UTP | * IoT Gateway dapat mengecek Cloud Platform *datalog command* dan dapat menyimpan kedalam IoT Gateway *datalog info* |