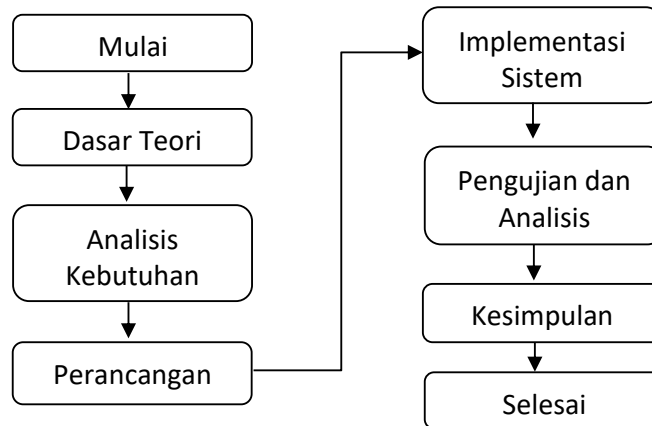


## BAB 3 METODOLOGI

Metodologi penelitian menjelaskan urutan langkah-langkah yang akan digunakan dalam penelitian ini. Dengan adanya sistematika penulisan ini, proses penelitian diharapkan dapat dipahami dengan baik.

Adapun urutan penelitian yang akan dilakukan adalah sebagai berikut:



### 3.1 Dasar Teori

Peneliti akan menggunakan *Internet of Things* sebagai objek penelitian dengan menggunakan NodeMCU. Dasar teori yang digunakan diambil dari referensi ilmiah, jurnal penelitian yang diterbitkan oleh institusi terpercaya, dan beberapa penelitian yang dilakukan sebelumnya.

### 3.2 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengetahui kebutuhan peneliti tentang pengujian yang akan dilakukan dan sesuai dengan tujuan penelitian yang ada.

#### 3.2.1 Kebutuhan Perangkat Lunak

- NodeMCU Flasher Master
- NodeMCU Firmware
- ESPLorer
- Python
- Android Studio
- Google Chrome

#### 3.2.2 Kebutuhan Perangkat Keras

Perangkat yang dibutuhkan dalam penelitian ini adalah sebuah laptop Lenovo G400s sebagai *Subscriber* dengan spesifikasi:

- Processor Intel Core i5-3230M @3.0 GHz
- Memori 4GB

- HDD SATA 1000GB
- Terpasang *driver* USB untuk NodeMCU

Dan ponsel pintar LG Nexus 5 untuk menjalankan aplikasi dan bertindak sebagai *Subscriber* dengan spesifikasi:

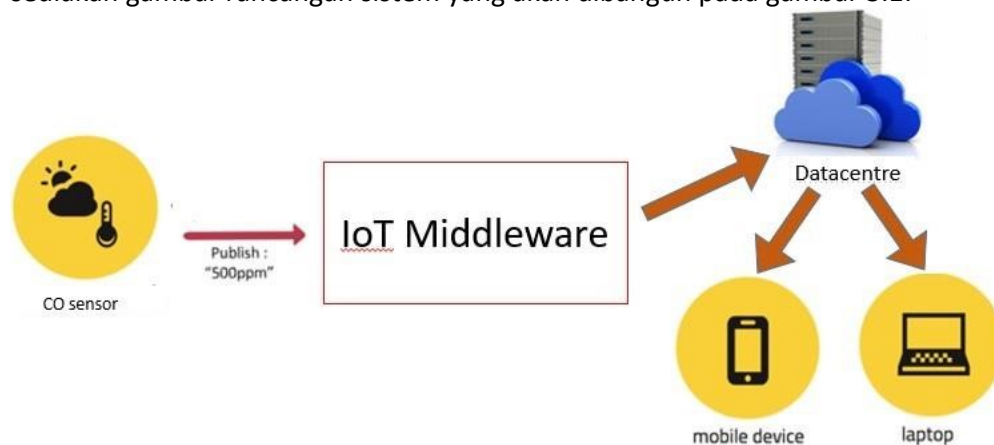
- Processor Qualcomm Snapdragon 800 @2.3GHz
- Memori 2GB
- Penyimpanan 16GB

Perangkat lain berupa 2 buah node sensor yang dibutuhkan dalam penelitian ini dan bertindak sebagai *Publisher*:

- 2 buah NodeMCU
- 2 buah Sensor MQ-7
- 2 buah Kabel micro USB
- 6 buah Kabel jumper

### 3.3 Perancangan Sistem

Peneliti akan mengembangkan sebuah sistem untuk mendeteksi kadar karbon monoksida dengan memanfaatkan IoT *middleware* berbasis event-driven yang sudah ada. Peneliti menggunakan sebuah sensor MQ7 pada setiap node. MQ7 merupakan sensor yang sensitif terhadap karbon monoksida (CO). Berikut peneliti sediakan gambar rancangan sistem yang akan dibangun pada gambar 3.1.



**Gambar 3.1 Rancangan sistem deteksi karbon monoksida**

*Middleware* bertugas mengirimkan data dari *publisher* (node sensor) ke *subscriber* (data center). IoT *middleware* yang digunakan merupakan *middleware* yang telah dikembangkan sebelumnya dan mendukung berbagai macam protokol komunikasi. Protokol komunikasi yang didukung adalah MQTT, CoAP, dan WebSocket.

Data dari *middleware* kemudian dikirim ke data center untuk diolah. Tujuan pengolahan data yaitu untuk mempermudah pengguna dalam memahami data yang diberikan dari sensor. Jika nilai pembacaan sensor melebihi batasan aman, maka data center akan mengirimkan status kadar CO meningkat dan berbahaya.

Namun jika nilai pembacaan sensor tidak melebihi batasan aman, maka data center akan mengirimkan status kadar CO di udara normal dan dalam keadaan aman. Data center juga bertugas menyediakan nilai rata – rata untuk ditampilkan pada aplikasi telepon pintar

Pada aplikasi web, pengguna dapat melihat data yang terdapat pada basis data berdasarkan topik yang di *subscribe*. Pada aplikasi telepon pintar, pengguna dapat menerima pemberitahuan apabila data hasil pembacaan nose sensor melebihi batas aman. Selain itu data juga bisa di tampilkan dalam nilai rata – rata setiap harinya.

### 3.4 Implementasi Sistem

Dari perancangan tersebut, sensor dihubungkan dengan NodeMCU yang memiliki integrasi dengan modul WiFi ESP8266, sehingga data hasil pembacaan sensor dapat langsung dikirim ke IoT *middleware* melalui jaringan WiFi. Bahasa Lua akan digunakan dalam memberikan perintah kepada NodeMCU. Perintah itu digunakan untuk menginisialisasi GPIO yang akan digunakan, beserta fungsi masing – masing GPIO yang digunakan. Perintah yang dituliskan dalam bahasa Lua juga digunakan untuk mengonfigurasi koneksi WiFi yang digunakan untuk mengirim data sensor ke *middleware*.

Pada *middleware*, peneliti dapat langsung mengirim data ke *middleware* yang sudah di konfigurasi sebelumnya. Data dari node sensor diterima oleh *middleware* dan di teruskan ke data center. Di data center, akan dijalankan logika, yaitu Jika nilai pembacaan sensor melebihi batasan aman, maka data center akan mengirimkan status kadar CO meningkat dan berbahaya. Namun jika nilai pembacaan sensor tidak melebihi batasan aman, maka data center akan mengirimkan status kadar CO di udara normal dan dalam keadaan aman. Data center juga bertugas menyediakan nilai rata – rata untuk aplikasi telepon pintar. Untuk mencapai itu semua, dibuat sebuah program menggunakan bahasa Python.

Aplikasi web berguna untuk menampilkan data sensor yang terdapat pada basis data. Aplikasi web di implementasikan menggunakan bahasa pemrograman HTML. Aplikasi telepon pintar berguna untuk memberikan pemberitahuan apabila data hasil pembacaan sensor melebihi batas aman. Selain itu aplikasi telepon pintar juga dapat menampilkan data terakhir dan menampilkan nilai rata – rata data yang terdapat pada basis data. Perintah tersebut akan dibuat menggunakan bahasa pemrograman Java. Adapun komponen pendukung yang digunakan adalah sebagai berikut:

- Bahasa pemrograman yang digunakan adalah Bahasa pemrograman Lua, sesuai dengan Bahasa pemrograman yang didukung oleh NodeMCU.
- *Middleware* yg digunakan merupakan *middleware* yang telah dikembangkan sebelumnya
- Untuk mengambil nilai rata -rata dari data yang ada di basis data pada data center, dibuat program menggunakan bahasa Python.

- Bahasa pemrograman yang digunakan untuk implementasi aplikasi telepon pintar adalah Java, sesuai bahasa pemrograman yang didukung oleh sistem operasi Android

### 3.5 Pengujian dan Analisis

Pengujian akan dilakukan pada sebuah ruangan berukuran 5×5 meter. Node sensor akan diletakkan di sebuah ruangan tersebut. Node sensor akan dinyalakan dan dibiarkan selama 60 detik untuk memanaskan sensor dan mendapatkan hasil pembacaan sensor yang akurat. Setelah 60 detik, node sensor akan mulai mengirim data ke *middleware* setiap 30 detik sekali. Pada ruangan pengujian, disediakan *middleware* yang telah dikembangkan sebelumnya oleh (Anwari, 2017). Node sensor akan mengirimkan data ke *middleware*, yang selanjutnya data dari *middleware* akan diteruskan ke data center untuk diolah dan disimpan. Data yang selesai diolah akan dikirimkan ke perangkat pengguna berupa laptop atau *smartphone* yang memiliki akses ke jaringan. Pengujian akan dilakukan dengan melakukan pengujian integrasi untuk menguji kebutuhan – kebutuhan fungsional yang akan dijelaskan pada bab selanjutnya. Sistem juga akan diuji apakah sesuai dengan kaidah komputasi berbasis konteks pada penelitian (Charith Perera, 2014). Pada pengiriman data dari sensor ke *middleware*, akan diuji kinerja sistem berdasarkan *Quality of Service* dengan parameter *delay*, *jitter* dan *throughput*.

### 3.6 Kesimpulan

Tahap akhir dari penelitian ini adalah pengambilan kesimpulan dan saran. Pengambilan kesimpulan dilakukan setelah proses perangkuman hasil analisis. Kesimpulan diharapkan dapat menjawab seluruh rumusan masalah yang telah disampaikan pada Bab I. Setelah itu, peneliti akan menerima saran dari hasil yang telah dicapai, yang akan digunakan untuk memperbaiki kesalahan pada pengembangan yang akan datang.