JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i1.5992

PROTOTYPE PENDETEKSI KEBAKARAN MULTIRUANG MENGGUNAKAN NODE MCU ESP8266 DENGAN NOTIFIKASI BOT TELEGRAM

Octavian Erwin Yudhistira[1*], Imam Suharjo[2]

[1,2]Universitas Mercu Buana Yogyakarta; Kampus 2, Depok, Sleman, Yogyakarta; 55283

Received: 2 Januari 2025
Accepted: 14 Januari 2025
Published: 20 Januari 2025

Keywords:
node MCU, Telegram

Corespondent Email:
octavianerwin25@gmail.com

Abstrak. Penelitian ini menyajikan pengembangan sistem deteksi kebakaran multiruangan yang terintegrasi dengan bot Telegram untuk notifikasi yang efisien dan tepat waktu. Sistem ini memanfaatkan NodeMCU ESP8266 sebagai *CPU*, yang memungkinkan koneksi beberapa sensor api, gas, dan suhu, untuk memantau berbagai ruangan. Setelah mendeteksi api, kebocoran gas, atau suhu melebihi 35°C, sistem memicu peringatan yang dikirimkan ke BOT Telegram. BOT ini akan mengirimkan notifikasi ke pengguna, memberikan pembaruan secara *real-time* tentang insiden kebakaran. Integrasi *BOT* Telegram meningkatkan keandalan sistem dan pengalaman pengguna dengan memberikan peringatan tepat waktu kepada pengguna di mana pun lokasi pengguna. Hasil eksperimen memvalidasi kemampuan sistem untuk mendeteksi kejadian kebakaran secara akurat dan segera memberi tahu pengguna melalui BOT Telegram, yang berpotensi membantu dalam respons cepat dan meminimalkan kerusakan terkait kebakaran.

Abstract. *This research presents the development of a multi-room fire detection system integrated with a Telegram bot for efficient and timely notifications. The system leverages the NodeMCU ESP8266 as the central processing unit, enabling the connection of multiple flame, gas, and temperature sensors, to monitor various rooms. Upon detecting a fire, gas leak,or temperature exceeds 35°C the system triggers an alert, which is transmitted to a Telegram bot. This bot, in turn, sends notifications to designated users, providing real-time updates on the fire incident. The integration of the Telegram bot enhances the system's reliability and user experience by delivering timely alerts to users regardless of their location. Experimental results validate the system's ability to accurately detect fire events and promptly notify users through the Telegram bot, potentially aiding in rapid response and minimizing fire-related damages.*

1. PENDAHULUAN

Kebakaran merupakan salah satu bencana yang sering terjadi dan menimbulkan kerugian yang sangat besar. Kebakaran terjadi ketika suatu zat mencapai suhu kritis dan bereaksi secara kimia dengan oksigen. yang menghasilkan panas, nyala api, cahaya, asap, uap air, karbon monoksida, karbon dioksida, atau produk dan efek lainnya [1]. Mainah, dkk. membuat sebuah penelitian dengan judul Rancang Alat Pendeteksi Kebakaran Dengan Sensor Asap Menggunakan Arduino Uno, dikarenakan pada tahun 2015 hingga 2016 di Jakarta sudah terjadi lebih dari 250 kasus kebakaran yang ditaksir kerugiannya mencapai Rp. 200 milyar [2]. Sistem deteksi kebakaran konvensional yang ada saat ini seringkali memiliki keterbatasan dalam mendeteksi kebakaran secara dini dan akurat. Oleh karena itu, penelitian ini bertujuan untuk merancang

dan membangun sebuah prototype sistem deteksi kebakaran berbasis *IoT* yang dapat mengatasi permasalahan tersebut. Penelitian ini dilakukan dengan tujuan mengurangi frekuensi terjadinya kebakaran dengan memanfaatkan pesatnya perkembangan teknologi saat ini [3]. Sistem ini dirancang untuk mencegah terjadinya kebakaran dengan mengirimkan data secara real-time dan memberikan notifikasi langsung melalui aplikasi [4].

ESP8266 adalah sebuah *chip* yang menggabungkan kemampuan mikrokontroler dengan konektivitas *Wi-Fi* tanpa modul tambahan. Alat yang sangat berguna untuk membuat proyek *IoT*.

Api adalah sesuatu yang dapat menghasilkan panas, cahaya, dan berbagai hasil reaksi kimia lainnya. Penemuan cara membuat api merupakan salah satu hal yang paling berguna bagi manusia, karena dengan api manusia bisa aman dari hewan buas, memasak makanan, dan mendapat sumber cahaya serta menjaga dirinya agar tetap hangat. Selain sesuatu yang berguna, api juga bisa membuat bencana yang bisa menyebabkan kerugian bagi manusia itu sendiri.

Internet of Things (*IoT*) adalah sebuah istilah dalam internet masa depan dimana benda-benda dapat berkomunikasi satu sama lain melalui sebuah jaringan internet [5]. Munculnya teknologi *Internet of Things* (*IoT*) menawarkan solusi yang menjanjikan dalam pengembangan sistem diberbagai bidang. Salah satu terobosan di dunia kesehatan dengan menggunakan platform *IoT* ialah penelitian yang dilakukan oleh Suzan, dkk. melakukan pengembangan sistem dibidang kesehatan dengan membuat penelitian dengan judul Rancang Bangun Pendeteksi Dehidrasi Manusia Berdasarkan Warna Urine Berbasis Internet Of Things (IoT) [6].

IoT memungkinkan interkoneksi berbagai perangkat dan sensor, sehingga memungkinkan pemantauan kondisi lingkungan secara *real-time* dan akurat. Dengan memanfaatkan sensor suhu, asap, dan api yang terhubung ke internet, sistem deteksi kebakaran berbasis *IoT* dapat mendeteksi tanda-tanda awal kebakaran dengan cepat. Data dari sensor yang di dapat selanjutnya dikirim melalui modul komunikasi ESP8266 sehingga pengguna ponsel berbasis Android dapat menerima data tersebut [7]. Selain itu, *IoT* juga memungkinkan integrasi dengan sistem alarm dan notifikasi, sehingga tindakan penyelamatan dapat dilakukan dengan segera.

Penelitian ini bertujuan untuk merancang, membangun, dan menguji kinerja sebuah prototype sistem deteksi kebakaran berbasis *IoT* yang dapat mendeteksi kebakaran secara dini dan akurat. Sistem ini akan dilengkapi dengan sensor suhu, asap, dan api yang terhubung ke platform *IoT*. Data yang diperoleh dari sensor akan diolah secara real-time untuk mendeteksi adanya potensi kebakaran. Penggunaan internet sebagai media komunikasi membuat jarak tidak terbatas [8]. Apabila terdeteksi kebakaran, sistem akan mengirimkan notifikasi kepada pengguna melalui perangkat mobile. Ruang lingkup penelitian ini meliputi pemilihan komponen *hardware* yang sesuai, pengembangan perangkat lunak untuk pengolahan data dan pengiriman notifikasi, serta pengujian kinerja sistem dalam berbagai kondisi simulasi kebakaran.

Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan teknologi deteksi kebakaran, khususnya dalam pemanfaatan teknologi *IoT*. Sistem deteksi kebakaran berbasis *IoT* yang dikembangkan dalam penelitian ini memiliki potensi untuk di aplikasikan pada berbagai jenis bangunan, seperti rumah tinggal, gedung perkantoran, pabrik, dan fasilitas umum lainnya. Selain itu, penelitian ini juga dapat menjadi referensi bagi penelitian selanjutnya dalam pengembangan sistem deteksi kebakaran yang lebih canggih dan terintegrasi.

Salindeho dan Wellem membuat penelitian dengan judul Perancangan Dan Implementasi Sistem Pendeteksi Dan Peringatan Kebakaran Berbasis Iot Menggunakan Nodemcu Esp8266 Dan Sensor API menggunakan sensor api, penelitian ini disempurnakan oleh penulis dengan menggunakan multi sensor dan multiruang [9]. Dengan adanya potensi pengembangan tersebut, penulis ingin membuat sebuah Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram untuk dapat menginformasikan gejala awal terjadinya kebakaran dan dapat dilihat melalui *smartphone* secara *real-time*. Sehingga mampu mengurangi ataupun meminimalisir kerugian yang disebabkan oleh musibah kebakaran serta

dapat mengurangi risiko terjadinya musibah kebakaran.

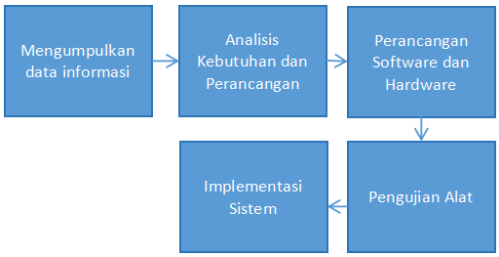
2. TINJAUAN PUSTAKA

Dengan kombinasi teknologi *IoT*, mikrokontroler Node MCU ESP8266, dan integrasi beberapa jenis sensor, penelitian ini diharapkan dapat menciptakan sistem pendeteksi kebakaran yang lebih responsif, efisien, dan dapat memonitor kondisi multiruang secara *realtime*. Penerapan telegram bot memberikan nilai tambah berupa kemudahan akses dan pengiriman notifikasi cepat kepada pengguna untuk meningkatkan kesiapsiagaan terhadap kebakaran. Penelitian ini menawarkan solusi inovatif yang belum banyak diterapkan dalam sistem konvensional.

3. METODE PENELITIAN

3.1. Tahap Pengerjaan Penelitian

Untuk membuat sistem deteksi kebakaran berbasis *internet of things* diperlukan beberapa tahapan, adapun tahapannya dapat dilihat pada gambar dibawah ini.

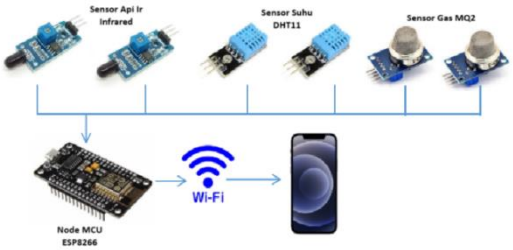


Gambar 1. Tahap Pengerjaan Penelitian

a. Tahap mengumpulkan data dan informasi
   Proses pengumpulan data dilakukan dengan cara mengkaji pustaka yang relevan. Hal ini bertujuan untuk memperoleh kerangka teoretis yang menjadi acuan dalam perancangan sistem deteksi kebakaran.
b. Tahap analisis kebutuhan dan perancangan
   Melalui analisis mendalam, penulis menentukan spesifikasi teknis yang diperlukan oleh sistem deteksi kebakaran. Selanjutnya, sebuah diagram blok sistem dirancang sebagai panduan dalam tahap implementasi.
c. Tahap perancangan *software* dan *hardware*
   Setelah merancang sistem, langkah berikutnya adalah perakitan perangkat keras. Sensor-sensor dihubungkan dengan NodeMCU ESP8266 sebagai langkah awal dalam membangun prototipe fisik. Proses ini kemudian dilanjutkan dengan penulisan kode program untuk mengontrol seluruh komponen sistem.
d. Tahap pengujian alat
   Tujuan dari tahap ini adalah untuk memastikan bahwa sistem deteksi kebakaran yang telah dibangun dapat beroperasi secara optimal dan handal dalam mendeteksi kebakaran.
e. Tahap implementasi sistem
   Melalui tahap ini, kita ingin memastikan bahwa sistem yang telah dibangun mampu mendeteksi kebakaran secara akurat dan memberikan respons yang cepat.

3.2. Gambaran Sistem

Sistem deteksi kebakaran berbasis *IoT* ini dirancang untuk meningkatkan keamanan dan mengurangi kerugian yang disebabkan oleh kebakaran, mengingat sifat kebakaran yang tidak dapat diprediksi dan dampaknya yang luas. Dengan memanfaatkan teknologi *IoT*, sistem ini diharapkan dapat memberikan kontribusi dalam upaya pencegahan dan penanggulangan kebakaran. Gambaran umum sistem pendeteksi kebakaran ini seperti pada gambar 2, setiap sensor (suhu, api, gas) pada setiap ruangan, akan mengirimkan informasi ke mikrokontroller NodeMCU ESP8266 yang terhubung pada *WiFi*, lalu dari mikrokontroller NodeMCU ESP8266 akan memproses data, dan hanya data yang memenuhi syarat berbahaya saja yang akan dikirim kepada pengguna melalui bot telegram.



Gambar 2. Gambaran Umum Sistem

3.3. Analisis Kebutuhan

Berdasarkan hasil analisis, Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram

memerlukan beberapa kebutuhan dalam perancangan sistem ini.

3.3.1. Kebutuhan Input

Berdasarkan kebutuhan *input* atau masukan dari Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram antara lain sebagai berikut:

a. Data sensor api.
b. Data sensor suhu.
c. Data sensor gas.

3.3.2. Kebutuhan Output

Berdasarkan hasil analisis, Kebutuhan *output* atau keluaran dari Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram antara lain sebagai berikut:

a. Notifikasi sensor api.
b. Notifikasi sensor suhu.
c. Notifikasi sensor gas.

3.3.3. Kebutuhan Hardware

a. Sensor suhu DHT22 2 buah
b. Sensor api Ir Infrared 2 buah
c. Sensor gas MQ2 2 buah
d. NodeMCU ESP8266 1 buah
e. Step Down LM2596 1 buah
f. Adaptor ADS25FSG-12 1 buah
g. Relay HK19F 1 buah
h. Kapasitor 104J100 1 buah
i. Resistor 0.25 watt 1 buah
j. Adapter Converter US to EU 1 buah
k. Transistor vector
l. Dioda 1 buah
m. Smartphone
n. Kabel Jumper male to male
o. Board 5cmx7cm
p. Tenol
q. Solder

3.3.4. Kebutuhan Software

Kebutuhan *software* atau perangkat lunak dari Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram antara lain sebagai berikut:

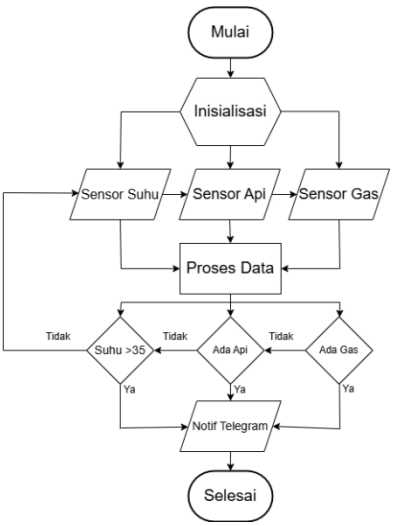
a. Arduino IDE 2.3.3
b. Telegram

3.4. Perancangan Sistem

Alat yang terdapat pada Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram akan melakukan pembacaan sensor-sensor yang diletakkan pada ruangan. Kemudian nilai dari pembacaan sensor tersebut akan ditampilkan pada notifikasi telegram apabila memenuhi kriteria tertentu. Oleh karena itu, perancangan Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram dapat melakukan pertukaran data secara *real-time*.

3.4.1. Perancangan Alur Kerja Sistem

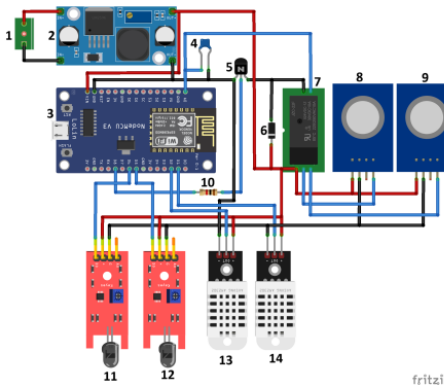
Gambar 3 menjelaskan tentang flowhart sistem yang mana nilai dari sensor suhu, sensor gas, dan api akan dibaca oleh NodeMCU ESP8266 dalam bentuk sinyal analog mapaun digital, serta apabila terdeteksi indikasi kebakaran, maka sistem akan mengirimkan notifikasi melalui bot telegram. Untuk sensor suhu, apabila terdeteksi lebih dari 35 derajat celcius, maka sistem akan mengirimkan notifikasi melalui bot telegram. Untuk sensor gas, apabila terdapat kebocoran gas atau terdapat asap, maka sistem akan mengirimkan notifikasi melalui bot telegram. Untuk sensor api, apabila terdapat api yang menyala, maka sistem akan mengirimkan notifikasi melalui *bot* telegram. Semua notifikasi tersebut akan dikirimkan secara *realtime*.



Gambar 3. Flowchart Sistem

3.4.2. Perancangan Komponen Hardware

Perancangan komponen *hardware* atau perangkat keras pada Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram terdiri dari Sensor suhu DHT22, sensor api Ir Infrared, sensor gas MQ2, NodeMCU ESP8266, step Down LM2596, adaptor ADS25FSG-12, relay HK19F, kapasitor 104J100, resistor 0.25 watt, adapter converter US to EU, transistor vector, Kabel jumper male to male, board 5cmx7cm. Terdapat power atau daya untuk menyalakan sistem, yang terhubung langsung ke NodeMCU ESP8266. Semua sensor akan terhubung, dan mengirimkan data ke NodeMCU ESP8266, lalu dari NodeMCU ESP8266 akan melakukan pemrosesan data. Apabila data yang dikirim melalui sensor ke NodeMCU ESP8266 memenuhi syarat tertentu, maka dari NodeMCU ESP8266 akan mengirimkan notifikasi ke BOT Telegram secara *real-time* guna dilakukan pemantauan oleh pengguna. Dapat dilihat untuk skema rangkaian sistem pada gambar 4.

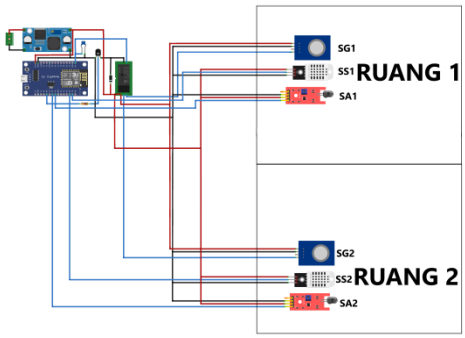


Gambar 4. Gambaran Rangkaian Sistem

Penjelasan gambar 4 sebagai berikut :

1. Adaptor ADS25FSG-12
2. Step Down LM2596
3. NodeMCU ESP8266
4. Kapasitor 104J100
5. Transistor vector
6. Dioda
7. Relay HK19F
8. Sensor gas MQ2 ruang 1
9. Sensor gas MQ2 ruang 2
10. Resistor 0.25 watt
11. Sensor api Ir Infrared ruang 1
12. Sensor api Ir Infrared ruang 2
13. Sensor suhu DHT22 ruang 1
14. Sensor suhu DHT22 ruang 2

Apabila diterapkan pada multiruang, berikut adalah sketsa multiruangnya :



Gambar 5. Gambaran Sketsa Multiruang

Penjelasan gambar 5 sebagai berikut :

1. SG1 adalah sensor gas ruang 1
2. SS1 adalah sensor suhu ruang 1
3. SA1 adalah sensor api ruang 1
4. SG2 adalah sensor gas ruang 2
5. SS2 adalah sensor suhu ruang 2
6. SA2 adalah sensor suhu ruang 2

Pada gambar 5 bisa dilihat bahwa setiap ruangan dilengkapi dengan 3 jenis sensor (sensor gas, sensor suhu, dan sensor api) yang terhubung pada 1 mikrokontroller.

3.4.3. Konfigurasi Sensor Suhu

Sensor suhu DHT22 menggunakan kombinasi komponen pengukuran kelembaban resistif dan komponen pengukuran suhu berupa *NTC* (*Negative Temperature Coefficient*). Ketika sensor dialiri listrik, komponen-komponen ini akan berubah resistansinya sesuai dengan perubahan suhu dan kelembaban di sekitarnya. Perubahan resistansi ini kemudian diubah menjadi sinyal digital yang dapat dibaca oleh mikrokontroler. Pada gambar 4 dan 5, digunakan 2 sensor suhu yang langsung terhubung ke NodeMCU ESP8266. Sensor suhu DHT22 memiliki 3 pin, yaitu pin VCC,

DATA, dan GND. Pin VCC digunakan untuk memberikan tegangan ke sensor. Pin ini dihubungkan ke mikrokontroller NodeMCU ESP8266 ke sumber tegangan positif. Pin DATA digunakan untuk mengirim dan menerima data digital dari sensor ke mikrokontroller NodeMCU ESP8266. Maka pin ini dihubungkan ke mikrokontroller NodeMCU ESP8266. Pin GND, atau *ground*. Pin ini digunakan untuk menghubungkan sensor ke *ground* atau massa. Pin ini bisa dihubungkan ke pin *ground* pada NodeMCU ESP8266 atau ke papan *board.*

3.4.4. Konfigurasi Sensor Api

Sensor api IR atau *infrared flame sensor* adalah komponen elektronik yang dirancang untuk mendeteksi keberadaan api atau panas yang intens dengan memanfaatkan radiasi inframerah yang dipancarkan oleh api.

Pada gambar 4 dan 5, digunakan 2 sensor api yang langsung terhubung ke NodeMCU ESP8266. Sensor api Ir Infrared memiliki 3 pin. Pin tersebut adalah pin DO, GND dan VCC. Untuk pin DO (*Digital Output*), bisa dihubungkan langsung dari sensor api ke mikrokontroller NodeMCU ESP8266. Untuk pin GND (*Ground*) bisa dihubungkan langsung ke mikrontroller NodeMCU ESP8266 atau ke papan *board*. Untuk pin VCC digunakan untuk memberikan daya ke sensor melalui mikrokontroller NodeMCU ESP8266.

3.4.5. Konfigurasi Sensor Gas

Sensor gas MQ-2 bekerja berdasarkan prinsip perubahan resistansi bahan semikonduktor ketika terpapar gas tertentu. Ketika sensor terpapar gas, resistansinya akan berubah. Perubahan resistansi ini kemudian diukur dan dikonversi menjadi sinyal listrik yang dapat diinterpretasikan oleh mikrokontroler. Sensor gas MQ-2 bisa mendeteksi berbagai macam gas. Gas yang bisa dideteksi oleh sensor gas MQ-2 antara lain, gas LPG, hidrogen, metana, karbon monoksida, alkohol, asap, dan propana.

Pada gambar 4 dan 5, digunakan 2 sensor gas yang langsung terhubung ke NodeMCU ESP8266. Sensor gas MQ-2 memiliki 4 pin. Pin VCC, GND, AO, dan DO. Pin VCC dihubungkan ke tegangan positif pada mikrokontroller NodeMCU ESP8266. Untuk pin GND, bisa dihubungkan ke *ground* pada bagian mikrokontroller NodeMCU ESP8266 maupun ke papan *board.* Pin AO (*Analog Output*) digunakan untuk memberikan tegangan analog yang sebanding dengan konsentrasi gas yang terdeteksi. Untuk pin DO (*Digital Output*) digunakan untuk memberikan sinyal digital yang menunjukkan adanya gas. Keluaran sensor MQ-2dapat berupa digital dan analog. Dalam penelitian ini, menggunakan output pada sensor berupa keluaran analog [10].

3.4.6. Konfigurasi Step Down LM2596

Step Down LM2596 adalah komponen elektronik yang digunakan untuk menurunkan tegangan *DC*. Dari adaptor ADS25FSG-12 lalu kabel disambungkan ke Step Down LM2596. Lalu dari *step down* dihubungkan menggunakan kabel menuju NodeMCU ESP8266.

3.4.7. Pembuatan Bot Telegram

Telegram telah memunculkan platform untuk pada para pengembang pihak ketiga dalam membuat akun BOT, akun ini merupakan akun telegram ayng dapat dijalankan pada sebuah program, seperti pengguna mengirimkan pesan ke BOT menggunakan perintah yang dapat dimengerti oleh BOT, sehingga dapat langsung dijawab pesan tersebut [11].

Bot telegram adalah akun otomatis di aplikasi pesan telegram yang dirancang untuk melakukan berbagai tugas. Membuat dan menggunakan bot telegram umumnya gratis. Akan berbayar apabila dengan skala yang besar. Untuk membuat bot telegram bisa mengakses telegram *link* telegram berikut *https://t.me/BotFather*.

Gambar 6. Gambaran Pembuatan Bot Telegram

Seperti pada gambar 6, setelah mengakses *link* telegram tersebut, lalu bisa diklik *start.* Setelah klik *start* akan muncul beberapa pilihan, bisa klik bagian */newbot.* Lalu setelah itu bisa memasukkan nama untuk bot yang akan dibuat, dan *username* dari bot yang dibuat dengan akhiran "bot". Apabila *username* masih tersedia, maka bot sudah jadi, dan bot akan mengirimkan token. Apabila *username* tidak tersedia, maka bisa memasukkan *username* lain.

3.4.8. Perancangan Pengujian Sistem

Pengujian Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram berfungsi untuk membuktikan bahwa setiap sensor dan NodeMCU ESP8266 dapat bekerja sesuai yang diharapkan dan bisa menampilkan notifikasi telegram apabila syarat terpenuhi secara *real-time*. Untuk proses ini dilakukan pengujian dengan cara membuat simulasi terjadinya kebakaran dengan memicu sensor menggunakan lilin dan korek api gas. Hasil pengujian akan dimasukkan ke dalam tabel hasil pengujian sistem. Dengan pengujian ini, bisa menjadi tolak ukur, bahwa setiap sensor bisa berjalan dengan baik atau tidak.

a. Sensor api diletakkan pada samping lilin yang menyala.
b. Sensor suhu diletakkan pada samping lilin yang menyala.
c. Sensor gas diberikan picuan berupa gas dari korek api gas.
d. Memastikan NodeMCU ESP8266 bisa terhubung dengan *WiFI* dan bisa memberikan output berupa notifikasi ke bot telegram dengan ruangan yang spesifik pada saat syarat terpenuhi.

4. HASIL DAN PEMBAHASAN

4.1. Hasil Implementasi

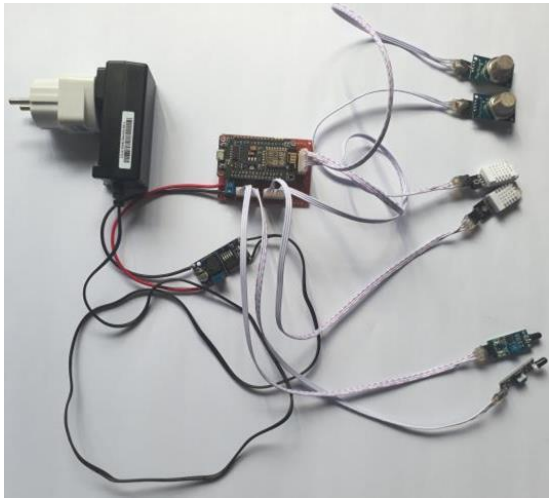
Pada bab ini akan dibahas hasil dari Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram. Beberapa hal yang akan dibahas adalah perangkat keras (*Hardware)*, perangkat lunak (*Software)*, pengujian sistem serta kekurangan dan kelebihannya.

4.1.1. Implementasi Perangkat Keras (Hardware)

Pada pembuatan Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram, penulis menggunakan mikrokontroller NodeMCU ESP8266, dengan menggunakan 3 sensor (api, suhu, dan gas) yang masing-masing sensor sebanyak 2 buah. Untuk sensor api menggunakan Ir Infrared, sensor suhu menggunakan DHT22, sensor gas menggunakan MQ2. Ketiga sensor tersebut terhubung langsung ke mikrokontroller NodeMCU ESP8266. Semua sensor tersebut akan mengirimkan sinyal digital maupun analog, yang nantinya akan dibaca oleh mikrokontroller NodeMCU ESP8266. Dari mikrokontroller NodeMCU ESP8266 akan memberikan sebuah seleksi, apabila syarat dan ketentuan terpenuhi maka mikrokontroller NodeMCU ESP8266 akan mengirimkan pesan melalui bot telegram melalui jaringan internet.

Dalam membuat Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram adalah mengkonfigurasi tiap sensor dengan NodeMCU ESP8266. Dikarenakan setiap sensor memiliki *output* 3.3V, maka diperlukan tambahan daya untuk menghidupkan seluruh sensor yang terhubung. Adaptor ADS25FSG-12 yang

terhubung ke stop kontak berfungsi untuk mengubah arus bolak-balik (*AC*) dari stopkontak listrik menjadi arus searah (*DC*) dengan tegangan dan arus tertentu. Lalu dari adaptor dihubungkan ke Step Down LM2596. Step Down LM2596 befungsi untuk menurunkan tegangan *DC* yang lebih tinggi menjadi tegangan *DC* yang lebih rendah. Dari Step Down LM2596 terdapat 2 output yaitu positif dan negatif. Untuk output positif dibagi menjadi dua jalur. Jalur pertama untuk tegangan positif dihubungkan ke NodeMCU ESP8266 pada pin VIN. Lalu pada jalur kedua, tegangan positif dihubungkan pada dioda, yang selanjutnya dihubungkan pada pin VCC di setiap sensor. Akan tetapi pada sensor gas sebelum disambungkan pada pin VCC, tegangan positif dihubungkan pada relay, baru dihubungkan pada pin VCC pada sensor gas. Untuk *output* negatif dari Step Down LM2596 juga dibagi menjadi 2 jalur, jalur yang pertama terhubung ke NodeMCU ESP8266 pada pin GND (*Ground*). Lalu untuk jalur kedua, terhubung ke kapasitor pada pin negatif, dan dari kapasitor dihubungkan ke transistor pada pin emitter. Lalu dari transistor, pin emitter dihubungkan pada pin GND (*Ground*) pada semua sensor kecuali sensor gas. Dari pin collector pada transistor, dihubungkan pada dioda dan dilanjutkan pada pin relay. Dari mikrokontroller NodeMCU ESP8266 terdapat 6 pin yang digunakan untuk melakukan koneksi data, yaitu pin D1, D2, D5, D6, D8 dan AO. Dari pin D1 pada mikrokontroller NodeMCU ESP8266 dihubungkan pada pin AO di sensor suhu DHT22 ruangan 1. Dari pin D2 pada mikrokontroller NodeMCU ESP8266 dihubungkan pada pin AO di sensor suhu DHT22 ruangan 2. Pada pin D5 pada mikrokontroller NodeMCU ESP8266 dihubungkan pada pin AO di sensor api ir Infrared ruangan 1. Pada pin D6 pada mikrokontroller NodeMCU ESP8266 dihubungkan pada pin AO di sensor api ir Infrared ruangan 2. Pada pin D8 pada NodeMCU ESP8266 dihubungkan pada resistor 0.25 watt, lalu dari resistor dihubungkan pada transistor pada pin collector. Kombinasi resistor dan transistor memberikan fleksibilitas dan kontrol yang lebih baik dalam mengelola aliran arus listrik. Lalu pin AO pada mikrokontroller NodeMCU ESP8266 dihubungkan pada kapasitor, dan dihubungkan pada relay. Lalu dari relay dihubungkan pada pin AO pada pin sensor gas MQ-2 ruang 1, dan pin relay lainnya dihubungkan pada pin AO pada pin sensor gas MQ-2 ruang 2. Pada Gambar 7 adalah hasil dari perakitan Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram.



Gambar 7. Hasil Perakitan Sistem

4.1.2. Implementasi Perangkat Lunak (Software)

Pada pembuatan Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram, penulis menggunakan 2 software. Yaitu Arduino IDE 2.3.3 dan telegram. Arduino IDE 2.3.3 digunakan untuk melakukan *compile* sintak yang sudah dibuat agar bisa diunggah pada mikrokontroller NodeMCU ESP8266. Untuk telegram digunakan untuk menampilkan notifikasi indikasi kebakaran apabila syarat terpenuhi.

4.2. Pembahasan Sintak Program

Sintak adalah kode program yang memerintahkan sistem, sehingga sistem bisa berjalan seperti yang diharapkan. Sebelum melakukan pemrograman, terlebih dahulu melakukan penginstalan *software* Arduino IDE. Setelah melakukan penginstalan, mikrokontroller dihubungkan pada laptop menggunakan kabel *USB*. Lalu melakukan pengunduhan *driver* untuk NodeMCU ESP8266 agar papan mikrokontroller terdeteksi. Setelah itu bisa mengunduh *library* Universal Telegram BOT, DHT Sensor Library 1.4.6 dan WifiClientSecure. Universal Telegram BOT digunakan untuk menghubungkan NodeMCU ESP8266 ke bot telegram, untuk DHT Sensor Library digunakan untuk mengambil data dari sensor suhu DHT22. Library WifiClientSecure digunakan untuk menghubungkan ESP8266 ke internet secara aman melalui protokol *HTTPS*.

```
#include <DHT.h>
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

#define DHTTYPE DHT22
#define DHT1_PIN D1
#define DHT2_PIN D2
#define FLAME1_PIN D5
#define FLAME2_PIN D6
#define MQ2_SWITCH D8
#define MQ2_ANALOG_PIN A0

DHT dht1(DHT1_PIN, DHTTYPE);
DHT dht2(DHT2_PIN, DHTTYPE);

const char* ssid = "M";
const char* password = "1234567890";
const char* botToken = "7690086061:AAHX_XXySyAba1tXOn47rEzDD0JgCd02Y90";
const char* chatID = "1260972555";

WiFiClientSecure secured_client;
UniversalTelegramBot bot(botToken, secured_client);

unsigned long previousMillis = 0;
unsigned long telegramMillis = 0;
unsigned long mq2Millis = 0;
const unsigned long interval = 1000;
const unsigned long telegramInterval = 1000;
boolean switchSensorMq2 = false;

boolean alarmSentRuangan1 = false;
boolean alarmSentRuangan2 = false;
float temp1 = 0.0f;
float temp2 = 0.0f;
boolean flame1Detected = false;
boolean flame2Detected = false;
int mq2_1 = 0;
int mq2_2 = 0;
const int setPointSuhu = 35;
const int setPointGas = 700;

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi connected");

  secured_client.setInsecure(); // Non-verifikasi sertifikat SSL

  dht1.begin();
  dht2.begin();

  pinMode(FLAME1_PIN, INPUT);
  pinMode(FLAME2_PIN, INPUT);
  pinMode(MQ2_SWITCH, OUTPUT);
  pinMode(MQ2_ANALOG_PIN, INPUT);

  digitalWrite(MQ2_SWITCH, LOW);

  bot.sendMessage(chatID, "Alat Ready.", "");
}

void loop() {
  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    switchSensorMq2 = !switchSensorMq2;
    if (!switchSensorMq2) {
      digitalWrite(MQ2_SWITCH, HIGH);
      delay(10);
      int value = analogRead(MQ2_ANALOG_PIN);
```

```
    mq2_1 = value;
  }
  if (switchSensorMq2) {
    digitalWrite(MQ2_SWITCH, LOW);
    delay(10);
    int value = analogRead(MQ2_ANALOG_PIN);
    mq2_2 = value;
  }

  temp1 = dht1.readTemperature();
  temp2 = dht2.readTemperature();
  flame1Detected = digitalRead(FLAME1_PIN) == LOW;
  flame2Detected = digitalRead(FLAME2_PIN) == LOW;

  Serial.println("=== Ruangan 1 ===");
  Serial.print("Suhu (DHT1): ");
  Serial.print(temp1);
  Serial.println(" °C");
  Serial.print("Flame (Flame1): ");
  Serial.println(flame1Detected ? "Api terdeteksi" : "Tidak ada api");
  Serial.print("Gas MQ-2 (Sensor 1): ");
  Serial.println(mq2_1);

  Serial.println("=== Ruangan 2 ===");
  Serial.print("Suhu (DHT2): ");
  Serial.print(temp2);
  Serial.println(" °C");
  Serial.print("Flame (Flame2): ");
  Serial.println(flame2Detected ? "Api terdeteksi" : "Tidak ada api");
  Serial.print("Gas MQ-2 (Sensor 2): ");
  Serial.println(mq2_2);

  if (temp1 > setPointSuhu && flame1Detected && mq2_1 > setPointGas && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Ruangan 1 dalam keadaan kritis!!!\n";
    message += "Suhu :"+String(temp1,1)+"°C\n";
    message += "Terdeteksi api\n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
  }
        else if (temp1 > setPointSuhu && flame1Detected && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Ruangan 1 Terdeteksi Api dan Suhu Tinggi \n";
    message += "Suhu :"+String(temp1,1)+"°C\n";
    message += "Terdeteksi api";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
        }
        else if (flame1Detected && mq2_1 > setPointGas && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Ruangan 1 Terdeteksi Api dan Kebocoran gas \n";
    message += "Terdeteksi api\n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
        }
        else if (temp1 > setPointSuhu && mq2_1 > setPointGas && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Ruangan 1 Terdeteksi Suhu Tinggi dan Kebocoran gas \n";
    message += "Suhu :"+String(temp1,1)+"°C\n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
        }
  else if (temp1 > setPointSuhu && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Suhu tinggi mencapai "+String(temp1,1)+"°C di Ruangan 1!";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
  }
  else if (flame1Detected && !alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya! Terdeteksi api di Ruangan 1!";
```

```
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
   }
  else if (mq2_1 > setPointGas &&
!alarmSentRuangan1) {
    String message = "ALARM: ⚠ Bahaya!
Terdeteksi kebocoran gas di Ruangan 1!";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan1 = true;
    telegramMillis = currentMillis;
   }

  if (temp2 > setPointSuhu &&
flame2Detected && mq2_2 > setPointGas
&& !alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Ruangan 2 dalam keadaan kritis!!! \n";
    message += "Suhu
:"+String(temp2,1)+"°C\n";
    message += "Terdeteksi api\n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
   }
        else if (temp2 > setPointSuhu &&
flame2Detected && !alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Ruangan 2 Terdeteksi Api dan Suhu Tinggi
\n";
    message += "Suhu
:"+String(temp2,2)+"°C\n";
    message += "Terdeteksi api";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
        }
        else if (flame2Detected && mq2_2 >
setPointGas && !alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Ruangan 2 Terdeteksi Api dan Kebocoran
gas \n";
    message += "Terdeteksi api \n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
        }
        else if (temp2 > setPointSuhu &&
mq2_2 > setPointGas &&
!alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Ruangan 2 Terdeteksi Suhu Tinggi dan
Kebocoran gas \n";
    message += "Suhu
:"+String(temp2,2)+"°C\n";
    message += "Terdeteksi kebocoran gas";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
        }
  else if (temp2 > setPointSuhu &&
!alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Suhu tinggi mencapai
"+String(temp2,1)+"°C di Ruangan 2!";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
   }
  else if (flame2Detected &&
!alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Terdeteksi api di Ruangan 2!";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
   }
  else if (mq2_2 > setPointGas &&
!alarmSentRuangan2) {
    String message = "ALARM: ⚠ Bahaya!
Terdeteksi kebocoran gas di Ruangan 2!";
    Serial.println(message);
    bot.sendMessage(chatID, message, "");
    alarmSentRuangan2 = true;
    telegramMillis = currentMillis;
   }
  }

  if (currentMillis - telegramMillis >=
telegramInterval) {
   alarmSentRuangan1 = false;
   alarmSentRuangan2 = false;
  }
}
```

Gambar 8. Sintaks Program

Penjelasan fungsi utama dari *library* yang digunakan adalah DHT.h digunakan untuk membaca data suhu dan kelembaban dari sensor DHT22. ESP8266WiFi.h digunakan untuk menghubungkan modul ESP8266 ke jaringan WiFi. WiFiClientSecure.h digunakan untuk komunikasi aman melalui *SSL/TLS*. UniversalTelegramBot.h digunakan untuk mengirim pesan melalui *API* bot telegram. Terdapat 6 sensor yang digunakan yaitu DHT1_PIN, DHT2_PIN digunakan untuk sensor suhu pada ruangan 1 dan ruangan 2. FLAME1_PIN dan FLAME2_PIN digunakan untuk sensor api pada ruangan 1 dan ruangan 2. MQ2_SWITCH digunakan untuk pengaktifan sensor gas secara bergantian pada ruangan 1 dan ruangan 2. MQ2_ANALOG_PIN digunakan untuk membaca nilai analog dari sensor MQ2. SSID dan password WiFi digunakan untuk menghubungkan ESP8266 ke jaringan lokal. Fungsi WiFi.begin() menginisialisasi koneksi *WiFi*, dan loop while digunakan untuk melakukan perulangan hingga koneksi berhasil. Untuk dht1.begin() dan dht2.begin() untuk menginisialisasi sensor DHT22. Sedangkan pinMode() digunakan untuk mengatur mode pin (*input/output*). SetPointSuhu untuk menentukan ambang suhu (35°C). setPointGas untuk menentukan ambang nilai gas dari sensor MQ-2 (700). alarmSentRuangan1 dan alarmSentRuangan2 digunakan untuk memastikan alarm tidak dikirim berulang kali.

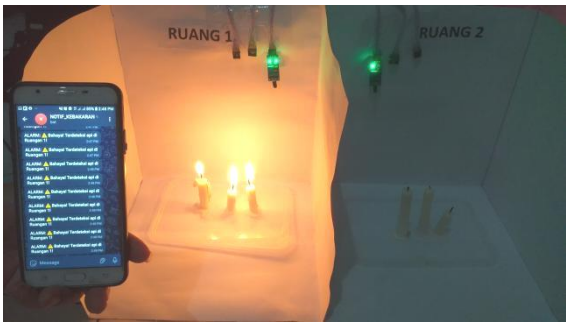
Jika suhu melebihi ambang batas, api terdeteksi, atau gas melebihi ambang batas, pesan dikirimkan ke telegram menggunakan bot. Alarm dikirim hanya sekali per kondisi bahaya. Apabila terdapat dua kondisi bahaya, maka (api dan gas, api dan suhu, suhu dan gas) alarm akan dikirimkan indikasi bahaya tersebut dengan detail. Jika semua kondisi bahaya (suhu tinggi, api, gas) terjadi bersamaan, alarm "kritis" akan dikirimkan dengan detail lengkap.

4.3. Pengujian Perangkat

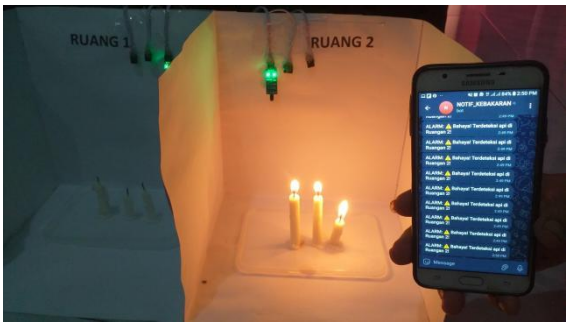
Fungsi dari pengujian perangkat adalah untuk memastikan bahwasannya Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram telah sesuai dengan yang diharapkan dan seluruh fungsi yang ada di dalam sistem dapat berjalan lancar sesuai dengan yang diharapan. Selain itu akan membahas kekurangan dan kelebihan dari sistem ini sehingga nantinya dapat menjadi bahan acuan untuk penelitian selanjutnya.

4.3.1. Pengujian Sensor Api

Sensor api adalah komponen elektronik yang dirancang khusus untuk mendeteksi adanya nyala api. Komponen ini sangat krusial dalam sistem keamanan, terutama dalam mendeteksi kebakaran sedini mungkin. Untuk melakukan pengujian terhadap sensor api penulis membuat simulasi kebakaran dengan membuat sumber api dari tiga lilin, meskipun pada kenyataannya jika terjadi kebakaran sumber api akan jauh lebih besar. Sensor api menggunakan inframerah dalam mendeteksi cahaya api sehingga semakin besar sumber api maka akan semakin jauh juga jarak deteksinya.



Gambar 9. Pengujian Sensor Api Ruang 1



Gambar 10. Pengujian Sensor Api Ruang 2

Tabel 1. Pengujian Sensor Api

No.	Jarak	Indikator Sensor	Notifikasi Telegram
1.	20cm	On	Ada
2.	40cm	On	Ada

3.	60cm	On	Ada
4.	80cm	On	Ada
5.	100cm	On	Ada
6.	120cm	Off	Tidak Ada

Tabel 2. Notifikasi Telegram Dari Sensor Api

No.	SA1	SA2	Notifikasi Telegram
1.	Tidak ada api	Tidak ada Api	-
2.	Ada api	Tidak ada Api	ALARM: ⚠ Bahaya! Terdeteksi api di Ruangan 1!
3.	Tidak ada api	Ada api	ALARM: ⚠ Bahaya! Terdeteksi api di Ruangan 2!
4.	Ada api	Ada api	ALARM: ⚠ Bahaya! Terdeteksi api di Ruangan 1! ALARM: ⚠ Bahaya! Terdeteksi api di Ruangan 2!

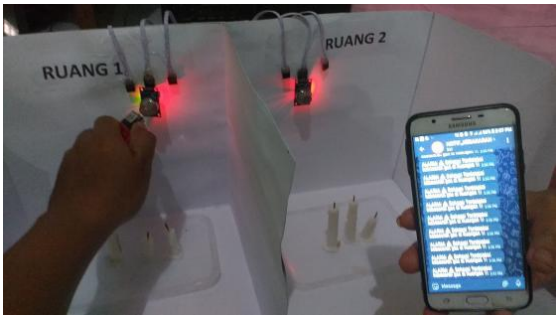
Gambar 9 dan 10 memperlihatkan bahwa sensor api pada sistem ini telah berhasil menjalankan fungsinya sesuai rancangan. Pada sistem ini, sensor api mampu mendeteksi api dari 3 lilin dari jarak 0cm hingga jarak 100cm. Untuk jarak lebih dari 100cm indikator sensor hanya berkedip, dan masih bisa mengirimkan notifikasi pada bot telegram. Bisa disimpulkan bahwa sensor api kesulitan mendeteksi api dari 3 lilin pada jarak lebih dari 100cm. Dan pada jarak 120cm, indikator pada sensor api sudah tidak menyala dan sudah tidak mengirimkan notifikasi telegram. Bisa disimpulkan bahwa sensor api sudah tidak mendeteksi nyala api dari 3 lilin.

Pada tabel 2, adalah detail notifikasi telegram dari berbagai kondisi. Berikut adalah beberapa kondisi yang sudah penulis ujicoba :

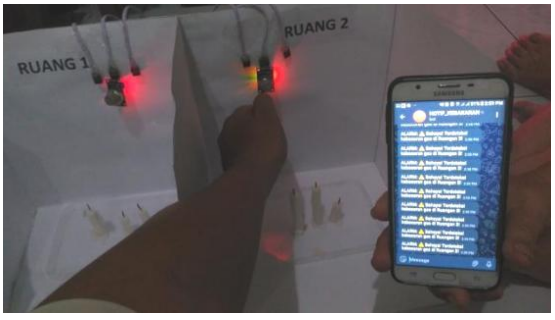
1. Kondisi no 1 adalah kondisi aman. Dimana tidak ada pacuan apapun dari sensor api pada ruang 1 dan ruang 2.
2. Kondisi no 2, sensor dipacu menggunakan lilin yang menyala pada ruang 1, dan sistem mengirimkan notifikasi terdeteksi api pada ruang 1 melalui bot telegram.
3. Untuk kondisi no 3 adalah kondisi dimana terdeteksi bahaya kebakaran pada ruang 2. Dimana pada ruang 2, sensor api ruang 2 dipacu menggunakan lilin yang menyala, dan sistem mengirimkan notifikasi terdeteksi api pada ruang 2 melalui bot telegram.
4. Dan untuk kondisi no 4, adalah kondisi dimana kedua sensor api dipacu menggunakan lilin yang menyala. Dan sistem mengirimkan notifikasi terdeteksi api, pada ruang 1 dan ruang 2.

4.3.2. Pengujian Sensor Gas

Sensor gas adalah komponen elektronik yang dirancang untuk mendeteksi keberadaan berbagai jenis gas di udara. Pada penelitian ini sensor gas dipacu menggunakan gas dari korek api gas (*butana*).



Gambar 11. Pengujian Sensor Gas Ruang 1



Gambar 12. Pengujian Sensor Gas Ruang 2

Tabel 3. Pengujian Sensor Gas

No.	Detik	Indikator Sensor	Notifikasi Telegram
1.	1 detik	Off	Tidak Ada
2.	2 detik	Off	Tidak ada
3.	3 detik	On	Ada
4.	4 detik	On	Ada
5.	5 detik	On	Ada

Tabel 4. Notifikasi Telegram Dari Sensor Gas

No.	SG1	SG2	Notifikasi Telegram
1.	Tidak ada gas	Tidak ada gas	-
2.	Ada gas	Tidak ada gas	ALARM: ⚠ Bahaya! Terdeteksi kebocoran gas di Ruangan 1!
3.	Tidak ada gas	Ada gas	ALARM: ⚠ Bahaya! Terdeteksi kebocoran gas di Ruangan 2!
4.	Ada gas	Ada gas	ALARM: ⚠ Bahaya! Terdeteksi kebocoran gas di Ruangan 1! ALARM: ⚠ Bahaya! Terdeteksi kebocoran gas di Ruangan 2!

Gambar 11 dan 12 memperlihatkan bahwa sensor gas pada sistem ini telah berhasil menjalankan fungsinya sesuai rancangan. Setelah berhasil mendeteksi gas dari korek api gas, sistem secara otomatis memberikan perintah untuk mengirimkan notifikasi melalui telegram seperti pada tabel 3 dan 4 bahwa terdapat indikasi kebakaran pada ruangan tertentu.

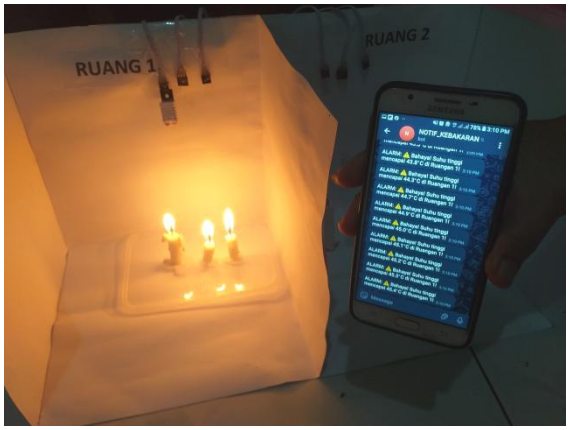
Pada tabel 3, sensor gas pada saat diberikan pacuan dari gas korek api, indikator sensor tidak langsung menyala, akan tetapi setelah 3 detik indikator baru menyala dan mengirimkan notifikasi bahaya melalui bot telegram. Memerlukan 3 detik untuk pemrosesan data bahwa terdapat kebocoran gas.

Pada tabel 4, peneliti sudah mencoba beberapa kondisi pada 2 ruangan tersebut, berikut adalah beberapa kondisi yang sudah penulis ujicoba :

1. Pada kondisi 1, adalah kondisi tidak berbahaya, dimana kedua sensor gas tidak diberi pacuan gas dari korek api, dan alat tidak mengirimkan notifikasi berbahaya ke bot telegram.
2. Pada kondisi ke 2, diberi pacuan gas hanya pada ruang 1, dan ruang 2 tidak diberi pacuan gas. Hasilnya sistem mengirimkan notifikasi kebocoran pada ruang 1.
3. Pada kondisi ke 3, hanya sensor gas ruang 2 yang diberi pacuan gas dari korek api, dan sistem mengirimkan notifikasi bahwa terjadi kebocoran gas pada ruang 2.
5. Pada kondisi ke 4, kedua sensor gas diberi pacuan gas dari korek api, dan hasilnya sistem mengirimkan notifikasi bahwa ruang 1 dan ruang 2 terdeteksi kebocoran gas.

4.3.3. Pengujian Sensor Suhu

Sensor suhu adalah perangkat elektronika yang dirancang untuk mendeteksi dan mengukur tingkat panas atau dingin suatu objek atau lingkungan. Sensor ini bekerja dengan mengubah energi panas menjadi sinyal listrik yang dapat dibaca oleh perangkat lain, seperti mikrokontroler atau komputer. Kegunaan sensor ini untuk mendeteksi suhu berlebih atau terdapat indikasi kebakaran dikarenakan suhu yang tinggi.



Gambar 13. Pengujian Sensor Suhu Ruang 1

Gambar 14. Pengujian Sensor Suhu Ruang 2

Tabel 5. Pengujian Sensor Suhu

No.	Waktu	Suhu	Notifikasi
1.	0 detik	27.5	Tidak ada
2.	5 detik	28.5	Tidak ada
3.	10 detik	30.2	Tidak ada
4.	15 detik	31.9	Tidak ada
5.	20 detik	33.7	Tidak ada
6.	25 detik	35.3	Ada
7.	30 detik	35.9	Ada
8.	35 detik	37.1	Ada

Tabel 6. Notifikasi Telegram Dari Sensor Suhu

No.	SS1	SS2	Notifikasi Telegram
1.	Suhu <35°C	Suhu <35°C	-
2.	Suhu >35°C	Suhu <35°C	ALARM: ⚠ Bahaya! Suhu tinggi mencapai 36.9°C di Ruangan 1!
3.	Suhu <35°C	Suhu >35°C	ALARM: ⚠ Bahaya! Suhu tinggi mencapai 44.1°C di Ruangan 2!
4.	Suhu >35°C	Suhu >35°C	ALARM: ⚠ Bahaya! Suhu tinggi mencapai 43.7°C di Ruangan 2!
			ALARM: ⚠ Bahaya! Suhu tinggi mencapai 38.2°C di Ruangan 1!

Gambar 13 dan 14 memperlihatkan bahwa sensor gas pada sistem ini telah berhasil menjalankan fungsinya sesuai rancangan.

Pada tabel 5 bisa dilihat bahwa sistem ini berhasil mendeteksi suhu melebihi 35°C dari suhu awal 27.5°C dalam kurun waktu 25 detik dengan jarak 5 cm dari titik api yang berasal dari lilin. Lalu sistem secara otomatis memberikan perintah untuk mengirimkan notifikasi melalui telegram dengan detail seperti pada tabel 6, bahwa ruangan tertentu terdapat indikasi kebakaran. Untuk penurunan suhu, dari 40°C ke suhu kurang dari 35°C membutuhkan waktu 16 detik. Lalu setelah suhu terdeteksi dibawah 35°C, sistem sudah tidak mengirimkan notifikasi indikasi kebakaran.

Pada tabel 6, peneliti sudah mencoba beberapa kondisi pada 2 ruangan tersebut, berikut adalah beberapa kondisi yang sudah penulis ujicoba :

1. Kondisi no 1, adalah kondisi yang tidak berbahaya. Kedua sensor tidak diberi pacuan apa-apa, dan dalam kondisi aman. Pada kondisi ini, sistem tidak mengirimkan notifikasi apapun.
2. Kondisi no 2, sensor suhu ruang 1 diberi pacuan panas dari lilin yang menyala, setelah 25 detik, sistem mengirimkan notifikasi suhu tinggi pada ruang 1.
3. Kondisi no 3, sensor suhu ruang 2 diberi pacuan panas dari lilin yang menyala, setelah 25 detik, sistem mengirimkan notifikasi suhu tinggi pada ruang 2.
4. Kondisi no 4, kedua sensor diberi pacuan panas dari lilin yang menyala, dan setelah 25 detik, sistem mengirimkan notifikasi

suhu tinggi pada ruang 1, dan suhu tinggi pada ruang 2.

4.3.4. Pengujian Multikondisi

Tabel 7. Pengujian Notifikasi Telegram dengan Multikondisi

No.	SA1 (Ada api?)	SS1 (Suhu tinggi?)	SG1 (Ada gas?)	SA2 (Ada api?)	SS2 (Ada suhu?)	SG2 (Ada gas?)	Notifikasi Telegram
1.	Tidak	Tidak	Tidak	Tidak	Tidak	Tidak	-
2.	Ya	Ya	Tidak	Tidak	Tidak	Tidak	ALARM: ⚠ Bahaya! Ruangan 1 Terdeteksi Api dan Suhu Tinggi Suhu :36.6°C Terdeteksi api
3.	Tidak	Ya	Ya	Tidak	Tidak	Tidak	ALARM: ⚠ Bahaya! Ruangan 1 Terdeteksi Suhu Tinggi dan Kebocoran gas Suhu :46.2°C Terdeteksi kebocoran gas
4.	Ya	Ya	Ya	Tidak	Tidak	Tidak	ALARM: ⚠ Bahaya! Ruangan 1 dalam keadaan kritis!!! Suhu :42.9°C Terdeteksi api Terdeteksi kebocoran gas
5.	Tidak	Tidak	Tidak	Ya	Ya	Tidak	ALARM: ⚠ Bahaya! Ruangan 2 Terdeteksi Api dan Suhu Tinggi Suhu :48.40°C Terdeteksi api
6.	Tidak	Tidak	Tidak	Tidak	Ya	Ya	ALARM: ⚠ Bahaya! Ruangan 2 Terdeteksi Suhu Tinggi dan Kebocoran gas Suhu :45.2°C Terdeteksi kebocoran gas
7.	Tidak	Tidak	Tidak	Ya	Ya	Ya	ALARM: ⚠ Bahaya! Ruangan 2 dalam keadaan kritis!!! Suhu :47.5°C Terdeteksi api Terdeteksi kebocoran gas

Pada tabel 7, bisa dilihat bahwa sistem ini sudah menunjang multikondisi indikasi kebakaran dalam 1 ruangan. Sistem ini bisa mendeteksi 2 kondisi bahaya, atau 3 kondisi bahaya sekaligus. Apabila sistem mendeteksi 3 kondisi berbahaya dalam 1 ruangan, maka sistem akan mengirimkan notifikasi "kritis" dengan detail yang lengkap. Apabila hanya terjadi 2 kondisi berbahaya dalam 1 ruangan, maka sistem mengirimkan 2 kondisi tersebut sekaligus.

4.4. Kelebihan dan Kekurangan Sistem

Pada Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram menggunakan kombinasi sensor suhu DHT22, sensor api, dan sensor gas MQ-2 untuk mendeteksi berbagai potensi bahaya.

4.4.1. Kelebihan Sistem

a. Bot telegram mengirimkan notifikasi langsung ke perangkat pengguna.

b. Dua ruangan dapat dipantau secara bersamaan.

c. Sensor Gas MQ-2 diaktifkan secara bergantian untuk menghemat daya.

4.4.2. Kekurangan Sistem

a. Belum ada sistem untuk mematikan alat secara manual seperti pada penelitian yang dilakukan oleh Wahyu Hidayat, dkk. yang berjudul Pembuatan Sistem Kebakaran Ruang Server Berbasis Web pada DISKOMINFO Kota Makassar yang sudah dilengkapi dengan saklar untuk menghidupkan atau mematikan alat [12].

b. Dapat dikembangkan untuk mendukung lebih banyak ruangan atau sensor tambahan.

5. DISKUSI

Penelitian ini menghasilkan sebuah prototipe pendeteksi kebakaran multiruang yang menggunakan NodeMCU ESP8266 sebagai pengendali utama dengan pemberitahuan secara

real-time melalui bot telegram. Sistem ini dirancang untuk mendeteksi kondisi berbahaya di dua ruangan yang berbeda dengan menggunakan sensor suhu (DHT22), sensor api (Ir Infrared), dan sensor gas (MQ2). Pengujian menunjukkan bahwa sistem mampu mendeteksi kondisi bahaya seperti kenaikan suhu, keberadaan api, dan kebocoran gas secara akurat. Pada sensor api, bisa mendeteksi api hingga jarak 100 cm hanya dengan menggunakan 3 lilin sebagai titik api. Pada sensor suhu, berhasil mendeteksi suhu melebihi 35°C dari suhu awal 27.5°C dalam kurun waktu 25 detik dengan jarak 5 cm dari titik api yang berasal dari 3 lilin. Dan pada sensor gas MQ2 bisa medeteksi konsentrasi gas dari korek api gas (*butana*).

Selain itu, fitur notifikasi telegram memungkinkan pengguna menerima informasi secara instan, sehingga memberikan waktu respons lebih cepat dalam menangani potensi kebakaran. Integrasi dengan *IoT* menjadikan sistem ini lebih fleksibel dibandingkan dengan sistem tradisional yang hanya menggunakan alarm lokal.

Jika dibandingkan dengan penelitian sebelumnya, keunggulan utama dari sistem ini adalah kemampuan memantau dua ruangan secara simultan dengan hanya menggunakan satu perangkat mikrokontroller NodeMCU ESP8266. Sistem ini juga memanfaatkan notifikasi telegram, yang menawarkan keunggulan komunikasi jarak jauh dan efisiensi biaya tanpa memerlukan aplikasi tambahan. Namun, terdapat kekurangan yang perlu diperhatikan. Seperti, stabilitas jaringan *WiFi* sangat memengaruhi performa sistem dalam mengirimkan notifikasi.

Dalam pengembangannya, sistem ini berpotensi untuk ditingkatkan dengan menambahkan redundansi komunikasi, seperti penggunaan jaringan seluler melalui modul *GSM*. Hal ini bertujuan untuk mengatasi permasalahan yang mungkin timbul akibat gangguan pada koneksi *WiFi*, terutama dalam situasi darurat di mana jaringan *WiFi* dapat terganggu atau terputus. Atau mungkin digunakan untuk menelepon atau mengirimkan sms ke pengguna apabila pengguna tidak memiliki data internet, maka notifikasi akan tetap sampai ke pengguna apabila terdapat indikasi kebakaran. Seperti pada penelitian yang dilakukan oleh Charly, dkk. dengan judul PERANCANGAN ALAT DETEKSI KEBAKARAN BERBASIS ARDUINO yang sudah dilengkapi dengan notifikasi *sms gateway*, sehingga pengguna tidak perlu khawatir apabila terdapat indikasi kebakaran pada saat kehabisan paket internet [13].

Terdapat perbedaan pada rangkaian sensor gas dibandingkan dengan penelitian sebelumnya. Pada penelitian ini sensor gas menggunakan pin analog *output* yang dihubungkan pada relay, sehingga konsumsi listrik lebih sedikit. Menggunakan pin analog memungkinkan peneliti mendapatkan data yang lebih akurat dan detail tentang konsentrasi gas. Dengan resolusi *ADC* yang cukup tinggi, peneliti dapat membedakan berbagai tingkat konsentrasi gas dengan lebih baik.

Dari hasil penelitian ini, terlihat bahwa prototipe pendeteksi kebakaran multiruang berbasis *IoT* dengan notifikasi telegram memiliki keunggulan dalam efisiensi dan kepraktisan penggunaan. Namun, pengembangan lebih lanjut diperlukan untuk meningkatkan keandalan sistem, baik dari sisi perangkat keras maupun perangkat lunak. Dengan implementasi yang lebih matang, sistem ini memiliki potensi untuk diterapkan pada skala yang lebih besar, seperti gedung perkantoran, pabrik, atau hunian, untuk meningkatkan keamanan dan mitigasi risiko kebakaran.

KESIMPULAN

Dengan melalui berbagai percobaan, dapat ditarik beberapa kesimpulan dari penelitian tersebut. Berikut ini adalah beberapa kesimpulan yang didapat peneliti :

1. Prototype mampu mendeteksi tiga parameter kebakaran, yaitu suhu, keberadaan api, dan konsentrasi gas, menggunakan sensor DHT22, flame sensor, dan sensor gas MQ-2.
2. Sistem secara otomatis mengirimkan notifikasi ke pengguna melalui bot telegram ketika parameter kebakaran melebihi ambang batas yang ditentukan, memberikan peringatan dini untuk potensi bahaya.
3. Mikrokontroller NodeMCU ESP8266 terbukti dapat terhubung dengan jaringan *Wi-Fi* secara stabil dan mengirimkan notifikasi rata-rata dalam waktu kurang dari 5 detik, menunjukkan responsivitas yang tinggi dalam situasi darurat.
4. Prototype ini dirancang untuk mendeteksi kebakaran di dua ruangan secara bersamaan, dengan kemampuan memonitor kondisi

setiap ruangan secara independen dan memberikan notifikasi spesifik sesuai lokasi bahaya.

5. Selain multiruang, prototype ini juga bisa mendeteksi indikasi kebakaran secara multikondisi.

UCAPAN TERIMA KASIH

Puji syukur dipanjatkan kepada Tuhan Yang Maha Esa, yang telah memberikan rahmat dan karunia-Nya, sehingga dapat diselesaikanya jurnal skripsi yang berjudul "Prototype Pendeteksi Kebakaran Multiruang Menggunakan Node MCU ESP8266 dengan Notifikasi BOT Telegram". Teruntuk ibu saya yang selalu memberikan semangat dan dukungan tanpa henti.

DAFTAR PUSTAKA

[1] A. Hartono, Siswanto, dan A. Widjaja, "Prototype Pendeteksi Kebakaran Menggunakan Sensor Flame, Sensor DHT11, dan Mikrokontroler NodeMCU ESP8266 Berbasis Website," Jakarta, 2022.

[2] M. Mainah, R. Getari, R. Fadillah, R. Restu, dan Ikhsan, "Rancang Alat Pendeteksi Kebakaran Dengan Sensor Asap Menggunakan Arduino Uno," *Jurnal Pustaka Robot Sister (Jurnal Pusat Akses Kajian Robotika, Sistem Tertanam, dan Sistem Terdistribusi)*, vol. 2, no. 1, hlm. 1–4, Jan 2024, doi: 10.55382/jurnalpustakarobotsister.v2i1.435.

[3] I. Dzikhrullah dan Z. Budiarso, "Rangkaian Rancang Bangun Alat Pendeteksi Kebakaran berdasarkan Asap dan Suhu pada Dapur Restoran Berbasis Arduino dan Internet of Things," *Jurnal Teknologi Informasi dan Komunikasi)*, vol. 7, no. 3, hlm. 2023, 2023, doi: 10.35870/jti.

[4] M. R. Fasya, Zaenudin, M. M. Efendi, dan L. D. Samsumar, "Implementasi Sistem Peringatan Dini Kebakaran Rumah Berbasis Internet of Things (IoT)," *Journal of Computer Science and Information Technology (JCSIT)*, vol. 1, no. 4, 2024.

[5] Siswanto, M. S. Laurin, dan D. W. Wibowo, "Prototype Akses Gedung Perpustakaan Dilengkapi Sistem Peringatan Dini Kebakaran Berbasis Internet of Things," *Jurnal PROSISKO*, vol. 9, no. 2, 2022.

[6] S. Zefi, M. M. Rose, dan D. T. Picara, "Rancang Bangun Pendeteksi Dehidrasi Manusia Berdasarkan Warna Urine Berbasis Internet of Things (IoT)," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, Okt 2024, doi: 10.23960/jitet.v12i3S1.5329.

[7] O. S. Giyanto dan Wahyat, "Prototype Alat Pendeteksi Kebakaran dengan Notifikasi Telegram dan Alarm Berbasis IoT," 2022.

[8] D. Halim dan D. V. S. Y. Sakti, "Sistem Peringatan Dini dengan Multi Sensor Berbasis Arduino untuk Monitoring Ruang Server," *SKANIKA*, vol. 4, no. 1, hlm. 1–6, 2021.

[9] G. G. Salindeho dan T. Wellem, "Perancangan dan Implementasi Sistem Pendeteksi dan Peringatan Kebakaran Berbasis IOT Menggunakan Nodemcu ESP8266 dan Sensor Api," *Jurnal Penerapan Teknologi Informasi dan Komunikasi*, vol. 02, no. 03, 2023.

[10] N. luh P. T. Ristanti dan R. Pradana, "Penggunaan Metode Threshold Dalam Pembuatan Sistem Pendeteksi Asap Dan Api Dengan Berbasis Firebase Dan Android Menggunakan Nodemcu Pada BJ House 77," *Technology of Information and Communication*, vol. 11, no. 1, 2022.

[11] T. Sulistyorini, N. Sofi, dan E. Sova, "Pemanfaatan Telegram untuk Pendeteksi Kebocoran Gas Berbasis NodeMCU," *JUIT*, vol. 3, no. 2, 2024.

[12] W. Hidayat, K. Imam, J. A. Safi', M. A. Leo, A. Faris, dan Al Faruq, "Pembuatan Sistem Kebakaran Ruang Server Berbasis Web pada DISKOMINFO Kota Makassar 1," *Progressive Information, Security, Computer, and Embedded System*, vol. 1, no. 1, hlm. 25–39, 2023, [Daring]. Tersedia pada: https://journal.diginus.id/index.php/PISCES/index

[13] C. J. Wungow, A. R. A. Ferdinandus, A. A. Adam, dan N. Komalig, "Perancangan Alat Deteksi Kebakaran Berbasis Arduino," *Jurnal SAINTEK*, vol. 04, no. 01, 2024.