

RANCANG BANGUN SISTEM PENDETEKSI KEBAKARAN BERBASIS MIKROKONTROLER ESP32

Bayu Kusumo¹, Teguh Ardiansyah²

^{1,2} Fakultas Teknik Universitas Krisnadwipayana

Bayu_kusumo@unkris.ac.id¹, maestrovng@gmail.com²

Abstrak - Kebakaran dapat dianggap sebagai satu diantara berbagai jenis musibah karena sifatnya yang membahayakan lingkungan dan masyarakat serta sulit dikendalikan. Kebakaran merupakan suatu insiden yang dapat membahayakan dan mengganggu kelangsungan hidup manusia, dan dapat terjadi di berbagai tempat seperti gedung, perumahan, dan hutan. Pada saat ini, orang – orang keluar bekerja dari pagi hingga sore hari meninggalkan rumah atau tempat tinggal cukup lama. Sehingga musibah seperti kebakaran kadang tak bisa diatasi saat meninggalkan rumah. Meskipun demikian, kebakaran dapat dihindari dengan berbagai cara. Oleh karena itu, dibutuhkan sistem yang mampu mendeteksi dan melakukan proteksi dini jika terjadi kebakaran. Sistem yang digunakan tersebut terdiri dari sensor api, asap dan suhu yang diintegrasikan dengan Internet of Things (IoT) menggunakan ESP32. Dimana sensor api dapat mendeteksi api hingga jarak 160 cm pada sudut 0° dan saat sudut 90° hanya pada jarak 12 cm. Sehingga semakin besar sudut antara detektor sensor api dengan sumber api, maka pembacaan sensor menjadi berkurang. Sensor asap mendeteksi asap saat kadar asap mencapai 402 PPM serta sensor suhu yang memberikan notifikasi ke android saat mendeteksi suhu melebihi 41.8 °C. Sensor suhu juga memiliki tingkat pembacaan yang sangat akurat karena dapat melakukan update yang lebih cepat dibandingkan dengan alat ukur suhu HTC-1. Sehingga saat terdeteksi adanya kebakaran, dapat langsung melakukan pencegahan dengan adanya notifikasi ke aplikasi android, alarm berupa buzzer dan memberikan proteksi menggunakan relay secara cepat dan efektif serta melakukan kontrol jendela menggunakan motor servo secara IoT.

Kata kunci : ESP32, *internet of things*, android, notifikasi, buzzer, relay, motor servo, sensor api, sensor suhu, sensor asap, kebakaran

Abstract - Fire can be considered as one of the various types of disasters due to its hazardous nature to the environment and society, as well as its challenging controllability. Fire is an incident that endangers and disrupts human survival and can occur in various places such as buildings, residences, and forests. Currently, people are often away from home or their living places for a significant amount of time, from morning until evening, for work. Therefore, disasters like fires might not be effectively addressed when leaving the premises. Nevertheless, fires can be prevented through various means. Thus, a system capable of early detection and protection is required in the event of a fire. The system in use comprises fire, smoke, and temperature sensors integrated with the Internet of Things (IoT) using ESP32. The fire sensor can detect fire up to a distance of 160 cm at a 0-degree angle, and at a 90-degree angle, only at a distance of 12 cm. Thus, as the angle between the fire sensor detector and the fire source increases, the sensor reading decreases. The smoke sensor detects smoke when the smoke concentration reaches 402 PPM, and the temperature sensor sends notifications to Android devices when the temperature exceeds 41.8°C. The temperature sensor also offers highly accurate readings, updating faster compared to the HTC-1 temperature measuring device. Consequently, when a fire is detected, immediate prevention can be initiated by sending notifications to the Android application, activating an alarm through a buzzer, providing rapid and effective protection using a relay, and controlling windows through IoT using a servo motor.

Keyword : ESP32, *Internet of Things*, Android, notification, buzzer, relay, servo motor, fire sensor, temperature sensor, smoke sensor, fire

1. PENDAHULUAN

Kebakaran dapat dianggap sebagai satu diantara berbagai jenis musibah karena sifatnya yang membahayakan lingkungan dan masyarakat serta sulit dikendalikan. Kebakaran merupakan suatu insiden yang dapat membahayakan dan mengganggu kelangsungan hidup manusia, dan dapat terjadi di berbagai tempat seperti gedung, perumahan, dan hutan. Kebakaran dapat disebabkan oleh banyak hal seperti membuang sisa rokok sembarangan, korsleting listrik, kebocoran gas, dan sebagainya, yang berpotensi menimbulkan kebakaran yang lebih besar lagi. Pada saat ini, orang – orang keluar bekerja dari pagi hingga sore hari meninggalkan rumah atau tempat tinggal cukup lama. Sehingga musibah seperti kebakaran kadang tak bisa diatasi saat meninggalkan rumah. Meskipun demikian, kebakaran dapat dihindari dengan berbagai cara. Dari situasi yang dijelaskan di atas, diperlukan suatu sistem yang dapat memberikan peringatan kebakaran kepada pemilik rumah atau petugas di perusahaan atau universitas yang terhubung, sehingga kejadian kebakaran dapat dideteksi dan ditangani dengan cepat dan efektif. Perancangan alarm dan sistem *monitoring* pendeteksi kebakaran berbasis *Internet of Things* (IoT) dengan notifikasi android merupakan solusi untuk memberikan notifikasi dini kepada pengguna jika terjadi kebakaran di suatu tempat sehingga pengguna dapat segera mengambil tindakan yang diperlukan. Mengingat saat ini, smartphone sudah menjadi kebutuhan pokok bagi masyarakat, salah satunya android. IoT memungkinkan perangkat untuk terhubung dan saling berkomunikasi melalui jaringan internet, sehingga solusi ini dapat memanfaatkan teknologi tersebut untuk menyajikan data yang efektif dan efisien. Dalam perancangan sistem ini, sensor-sensor yang dipasang pada lokasi tertentu akan memantau kondisi suhu. Jika suhu melebihi ambang batas yang telah ditentukan, sensor

akan mengirimkan sinyal ke perangkat *gateway* atau hub yang terhubung ke jaringan internet. Perangkat *gateway* atau *hub* akan meneruskan informasi ke platform *cloud* yang akan memproses data dan mengirimkan notifikasi ke aplikasi Android yang terhubung ke sistem. Dengan adanya solusi tersebut, diharapkan mampu memberi kenyamanan dan keefektifan bagi konsumen dalam memonitor kondisi lingkungan sekitar. Selain itu, solusi tersebut juga dapat terintegrasi dengan tindakan dini oleh pengguna jika terjadi kebakaran, seperti pemasangan sistem penyempnot air secara otomatis.

2. LANDASAN TEORI

2.1 Arduino

Arduino adalah sebuah papan sirkuit mikrokontroler yang dirancang sebagai platform terbuka untuk elektronik yang mudah digunakan di berbagai bidang. Chip mikrokontroler Atmel AVR digunakan sebagai prosesor arduino. Arduino mempunyai chipset Atmel AVR pada perangkat kerasnya dan perangkat lunak yang menggunakan bahasa pemrograman serta dikembangkan secara internal.



Gambar 2.1 Arduino Uno SMD CH340

Dalam dunia elektronik, arduino menjadi salah satu platform yang terkemuka di kalangan orang-orang yang baru belajar atau masih pemula. Salah satu keunggulan arduino adalah sifatnya yang dapat dimodifikasi oleh pengguna, sehingga memungkinkan individu manapun mampu berkreasi menciptakan perangkat kreatif interaktif dengan komponen fisik dan program yang ringkas dan fleksibel. Bahasa

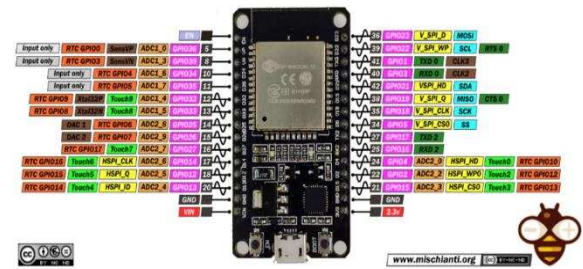
pemrograman arduino digunakan untuk memprogram mikrokontroler pada board arduino dan memiliki struktur syntax yang serupa dengan bahasa pemrograman C. Selain itu, komponen fisik dan program arduino didesain agar lebih mudah dipahami, sehingga dapat diakses oleh siapa saja yang tertarik. Dalam pengoperasiannya, varian mikrokontroler ATmega digunakan oleh Arduino yang diproduksi oleh Atmel, namun kini sudah tersedia kloning Arduino yang dibuat oleh individu atau perusahaan menggunakan jenis mikrokontroler yang berbeda tetapi masih dapat berjalan dengan perangkat lunak pengembangan Arduino IDE. Hal ini dilakukan untuk memberikan kemampuan menyesuaikan diri dalam pemilihan mikrokontroler. Selain itu, pengguna juga dapat mengunggah program baru ke board arduino dengan mudah melalui kabel konektor USB, tanpa memerlukan perangkat keras terpisah. Arduino memiliki kemampuan untuk berinteraksi dengan berbagai jenis komponen elektronik, termasuk speaker, jaringan komunikasi, tombol, motor, GPS, alat perekam gambar, dan LED, bahkan perangkat seperti gawai pintar atau layer lebar. Kemudahan dalam belajar menggunakan arduino diperkuat dengan kenyataan bahwa program arduino tersedia untuk diunduh tanpa biaya dan harga komponen yang ekonomis. Dikarenakan bersifat terbuka, arduino memiliki kelompok konsumen yang cukup banyak dan beragam serta berkontribusi untuk mengembangkan *scripting* dan library untuk menciptakan banyak macam aplikasi yang berkulat arduino.[10]

2.2 ESP32

Espressif System memperkenalkan ESP32 sebagai pengganti mikrokontroler sebelumnya, yaitu ESP8266. Kelebihan utama dari ESP32 adalah integrasi modul Wifi dan Bluetooth yang memudahkan pembuatan sistem IoT yang memerlukan koneksi nirkabel. Dalam hal ini, ESP32

menawarkan peningkatan dari ESP8266 karena fitur-fitur ini tidak tersedia di dalam ESP8266. Ada sebagian besar perangkat lunak yang memiliki kemampuan untuk memprogram ESP32, di antaranya adalah:

1. Arduino IDE
2. *Espressif IoT Development Framework*
3. Arduino Promini
4. *Ekstensi ESP-IDF Visual Studio Code*



Gambar 2.2 Pinout dari ESP32

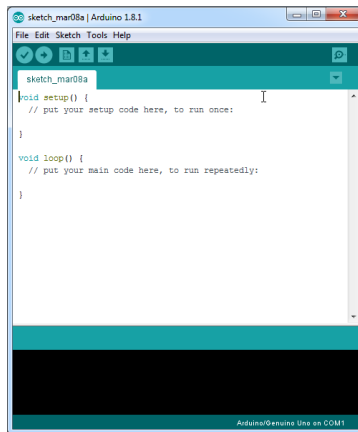
Mikrokontroler ESP32 dapat diimplementasikan dalam beberapa aplikasi, seperti:

1. Sistem pengontrol tanaman pintar (*Smart Garden*), yang dapat membantu dalam memantau kondisi pertumbuhan tanaman. Sistem ini memiliki sensor kelembaban udara, suhu, ketinggian air, dan kelembaban tanah, yang digunakan untuk memonitor kondisi tanaman secara akurat.
2. Sistem Kulkas Pintar (*Smart Fridge*), yang dapat mendeteksi secara otomatis bahan makanan yang tidak layak dikonsumsi, memberikan perkiraan tanggal kadaluwarsa bahan makanan, dan melacak kebiasaan penggunaan bahan makanan.[11]

2.3 Arduino IDE

Arduino dirancang untuk pemula atau bahkan untuk orang yang tidak memiliki pengetahuan dasar dalam pemrograman, karena menggunakan bahasa C++ yang telah disederhanakan melalui library. Arduino menggunakan perangkat lunak bernama Processing yang dirancang khusus

untuk menulis program yang dijalankan pada papan Arduino. Processing adalah bahasa pemrograman yang menggabungkan elemen-elemen dari bahasa C++ dan Java.



Gambar 2.3 Tampilan Arduino IDE

Perangkat lunak Arduino IDE dapat diinstal pada berbagai sistem operasi (OS) termasuk Windows, LINUX, dan Mac OS. Arduino IDE bukan hanya sekadar alat pengembangan, melainkan merupakan kombinasi dari perangkat keras (hardware), bahasa pemrograman, dan Integrated Development Environment (IDE) yang canggih. Integrated Development Environment (IDE) adalah perangkat lunak yang memainkan peran penting dalam menulis program, mengompilasi program menjadi kode biner, dan mengunggahnya ke memori mikrokontroler. Software IDE Arduino IDE terdiri dari tiga komponen utama, yaitu:

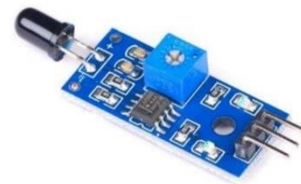
1. Bagian pertama dari Software IDE Arduino adalah editor program yang digunakan untuk menulis dan mengedit program menggunakan bahasa pemrograman Processing. Dalam konteks Arduino, program yang ditulis disebut sebagai sketch.
2. Komponen kedua dari Software IDE Arduino adalah compiler, yaitu modul yang bertugas mengonversi bahasa pemrograman Processing (kode program) menjadi kode biner. Kode biner adalah

satu-satunya format bahasa pemrograman yang dapat dipahami oleh mikrokontroler.

3. Bagian ketiga dari Software IDE Arduino adalah uploader, yaitu modul yang bertugas memasukkan kode biner ke dalam memori mikrokontroler. Modul ini bertanggung jawab untuk mengirimkan program yang telah dikompilasi ke mikrokontroler sehingga dapat dijalankan.[1]

2.4 Sensor Api IR Flame

Sensor IR Flame adalah sensor *infrared* yang berperan penting dalam mengindikasikan keberadaan api secara cepat. Sensor IR Flame menggunakan teknologi *infrared* untuk mengindikasikan keberadaan api dengan jangkauan panjang gelombang mulai dari 760 nm sampai 1.100 nm.



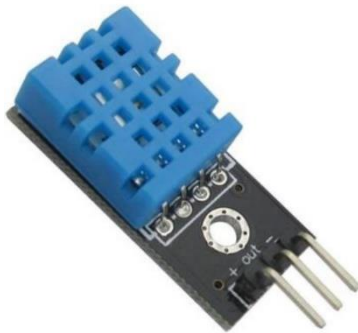
Gambar 2.4 Sensor Api

Terdapat 3 pin pada Sensor IR Flame, yakni diantaranya GND, Digital Output, dan VCC. Sensor tersebut berguna untuk mendeteksi adanya api dan dimanfaatkan untuk mengindikasikan titik sumber api saat kebakaran terjadi.

2.5 Sensor Kelembaban dan Suhu DHT11

Sensor ini merupakan sensor digital yang mampu mendeteksi temperatur dan kandungan air udara. Detektor atau sensor ini menyediakan data kelembaban dan suhu secara berurutan dengan menggunakan kabel penghubung. DHT11 dapat memberikan nilai kelembaban relatif dalam bentuk persentase dan nilai suhu dalam

derajat Celsius. Untuk mengukur kelembapan udara, DHT11 menggunakan komponen pengukuran resistif, sedangkan untuk mengukur suhu udara, DHT11 menggunakan komponen pengukuran NTC. Sensor jenis DHT11 digunakan sebagai detektor temperature dan dapat membaca temperatur udara pada saat kebakaran terjadi. Sensor DHT11 memiliki empat kaki pin, yaitu VCC, DATA, GND, dan satu pin yang tidak terpakai. [6]



2.5 Sensor DHT11

2.6 Sensor Gas MQ-2

Sensor berjenis MQ-2 ini dikenal sebagai perangkat yang digunakan guna mengetahui kebocoran beberapa jenis gas di lingkungan pertokoan atau perumahan. Sensor ini dapat digunakan untuk mendeteksi jenis gas seperti hidrogen, gas tabung, metana, karbon monoksida, dan Propana.



Gambar 2.6 Sensor MQ-2

Dalam Sensor MQ-2, terdapat empat pin penting yang terdiri dari GND, Analog Output, VCC, dan Digital Output. Selain itu, sensitivitas sensor dapat disesuaikan dengan menggunakan potensiometer. Sensor MQ-2 Berperan penting dalam

mengungkap adanya asap yang dihasilkan dari kebakaran.[7]

2.7 Relay

Relay merupakan suatu piranti yang digunakan sebagai semacam pemutus atau penghubung arus untuk mengontrol piranti lain. Relay dapat dikendalikan dari tegangan pin arduino sehingga dapat berfungsi saklar .



Gambar 2.7 Relay 2 channel

Terdapat tiga koneksi utama pada relay, yakni COM sebagai masukan dari piranti lain, NC (Normally Closed) pada saat kondisi normal terkoneksi ke kaki NC, dan NO (Normally Open) pada saat kondisi normal tidak terkoneksi, tetapi ketika relay menerima tegangan dari mikrokontroler mengakibatkan COM akan beralih dari NC ke NO.[8]

2.8 Motor Servo

Motor servo adalah sebuah motor dengan sistem *closed feedback* di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor.



Gambar 2.8 Motor Servo

Motor servo adalah motor yang berputar lambat, dimana biasanya ditunjukkan oleh rate putarannya yang lambat, namun demikian memiliki torsi yang kuat karena internal gearnya. Lebih dalam dapat digambarkan bahwa sebuah motor servo memiliki:

1. Tiga jalur kabel: power, ground, dan control
2. Sinyal kontrol mengendalikan posisi
3. Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar ± 20 ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari range sudut maksimum.
4. Konstruksi didalamnya meliputi internal gear, potensiometer, dan *feedback control*. [3]

2.9 Buzzer

Buzzer adalah sebuah komponen yang memiliki fungsi mengubah arus listrik menjadi suara dan pada dasarnya prinsip kerja buzzer hampir sama dengan speaker. Buzzer terdiri dari sebuah diafragma yang memiliki kumparan. Ketika kumparan tersebut dialiri arus listrik sehingga menjadi electromagnet, kumparan akan tertarik kedalam atau keluar tergantung dari polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap getaran diafragma secara bolak – balik sehingga membuat udara bergetar dan menghasilkan suara. Buzzer ini akan digunakan sebagai indicator apabila stang motor dipaksa lurus pada saat stang sepeda motor dikunci. [4]



Gambar 2.9 Buzzer

2.10 Android

Android merupakan sebuah program yang dimanfaatkan pada gawai dan terdiri dari sistem kernel, program perantara, dan aplikasi pengaman. Untuk menjadi pengembang pada platform android, java merupakan satu diantara banyak bahasa pemrograman yang diimplementasikan oleh sebagian banyak pengembang. Android memiliki aplikasi dasar seperti SMS, kontak, kalender, peta, email, browser, dan beberapa aplikasi lainnya. Dengan menyediakan suatu platform pengembangan yang dapat diakses tanpa batas, para *developer* aplikasi android mampu menciptakan aplikasi yang sangat inovatif dan beragam. Para pengembang Android diberi kebebasan untuk menggunakan piranti keras, menjalankan proses di latar belakang, memperoleh titik lokasi, menjadwalkan pengingat waktu, menampilkan notifikasi di bilah status, dan sebagainya. Hal ini dimungkinkan karena Android menyediakan platform pengembangan yang terbuka dan fleksibel. Versi Linux 2.6 digunakan oleh Android sebagai sistem inti yang bertanggung jawab untuk layanan seperti keamanan, stack jaringan, manajemen proses, manajemen memori, dan emulator perangkat keras. Kernel atau inti sistem juga berperan seperti layer abstraksi antara perangkat keras dan semua lapisan perangkat lunak. Android dilengkapi dengan banyak keunggulan dalam bentuk *software open source*, yang memberikan pengguna kebebasan dalam berkreasi untuk menciptakan aplikasi internal baru. Selain itu, Android juga menyediakan aplikasi produk asli google

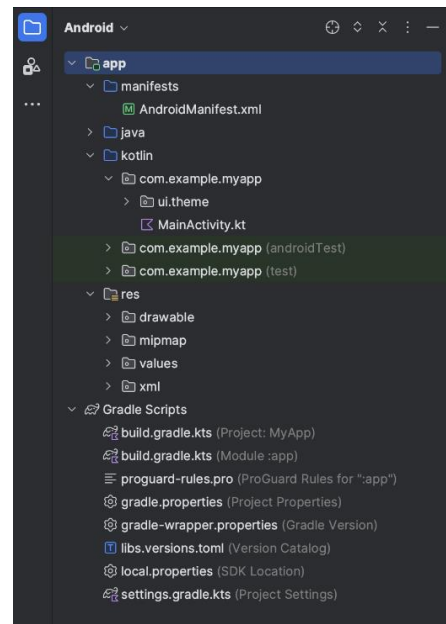
yakni google maps, email google, dan kalender google yang terintegrasi dengan baik. Banyak komunitas dibentuk yang terdiri dari peminat open-source untuk mengembangkan dan mengunggah *firmware* Android agar dapat disesuaikan dengan kebutuhan dan dilengkapi fitur baru seperti kemampuan dalam menampung aplikasi di memori eksternal. Mereka rutin memperbarui *firmware* kit dan mengintegrasikan fitur sistem yang belum dirilis secara resmi oleh otoritas android.[9]

2.11 Android Studio

IntelliJ IDEA merupakan developer andal yang membuat kode editor bernama Android Studio yang berbasis Integrated Development Environment untuk pengembangan aplikasi android. Android studio mempunyai berbagai fasilitas yang dapat memajukan kreativitas dan produktivitas dalam menciptakan aplikasi Android, seperti:

1. Sebuah sistem build yang fleksibel berbasis Gradle.
2. Memiliki banyak fitur serta kecepatan emulator yang cukup tinggi.
3. Platform integrasi yang menyediakan lingkungan yang terpadu di mana Anda dapat mengembangkan aplikasi yang kompatibel dengan berbagai perangkat. Android.
4. Fitur Edit Live yang memungkinkan pembaruan komponen secara real-time pada emulator dan perangkat fisik.
5. Template kode dan integrasi GitHub yang dapat membantu dalam pembuatan fitur aplikasi umum serta mengimpor kode sampel.
6. Alat pengujian dan framework yang komprehensif.
7. Alat lint yang digunakan untuk memantau kinerja, fungsi, kompatibilitas versi, dan masalah lainnya.
8. Support NDK dan C++.
9. Support untuk Google Cloud Platform, yang menyederhanakan integrasi

dengan Google Cloud Messaging dan App Engine.[5]



Gambar 2.1 Tampilan project android studio

2.12 Firebase

Firebase adalah platform database aplikasi realtime. Ketika terjadi perubahan data, aplikasi yang terhubung dengan Firebase akan secara otomatis melakukan pembaruan secara langsung di setiap perangkat yang terhubung, baik itu melalui website maupun aplikasi mobile. Firebase mempunyai library (pustaka) yang lengkap untuk sebagian besar platform web dan mobile. Firebase juga memiliki kemampuan untuk terintegrasi dengan berbagai framework lain seperti java, C++, javascript dan lain-lain. API (*Application Programming Interface*) digunakan sebagai penyimpanan dan sinkronisasi data dimana API akan mengambil data dalam format JSON (JavaScript Object Notation) dan menyimpannya di cloud. Data ini akan diperbarui secara otomatis dan sinkronisasi akan terjadi secara *real-time*. Firebase juga menyediakan beberapa fitur yaitu sebagai berikut.

1. Melalui fitur *Analytics*, Firebase dapat memonitor aktivitas pengguna dalam

menggunakan aplikasi dan menampilkan data tersebut dalam satu dashboard.

2. Pada bagian Develop, Firebase menyediakan berbagai fitur yang terdiri dari *authentication*, *crash reporting*, *storage*, realtime database, testlab, cloud messaging, dan hosting.

3. Dalam fitur *Grow*, Firebase dapat merilis sebuah produk aplikasi.

Firebase Realtime Database adalah platform penyimpanan data online yang yang memungkinkan aplikasi untuk menyimpan data secara efisien. Data disimpan dalam format JSON dan dapat disinkronkan secara langsung ke setiap perangkat yang terhubung. Layanan ini menawarkan tiga kemampuan utama, yaitu:

1. Fitur realtime, memungkinkan perubahan data pada database untuk segera disampaikan kepada semua klien yang terhubung secara otomatis dan cepat.

2. Fitur offline memungkinkan aplikasi tetap responsif bahkan ketika beroperasi tanpa sinyal internet. Hal ini dapat terjadi dikarenakan Firebase SDK yang mempunyai fungsi untuk menyimpan dan melacak perubahan data pada media penyimpanan pada perangkat pengguna. Saat perangkat terhubung ke internet lagi, Firebase SDK secara otomatis akan menyesuaikan data perubahan yang disimpan di media penyimpanan dengan kondisi terbaru dari server Firebase.

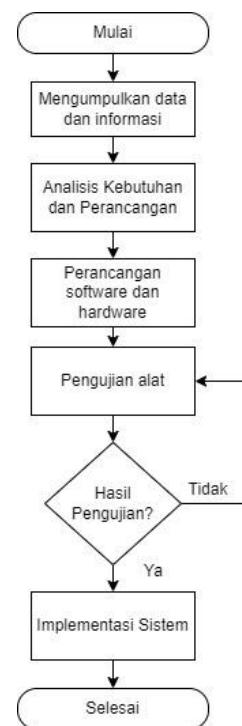
3. Accessible from client devices. Firebase Realtime Database menyediakan kemudahan untuk mengaksesnya secara langsung dari perangkat mobile atau browser web tanpa memerlukan server aplikasi.

Firebase memiliki database yang termasuk dalam kategori non-relasional atau NoSQL, yang berarti database ini tidak menggunakan struktur tabel dalam implementasinya. Selain itu, data dalam database Firebase tidak disimpan secara lokal pada perangkat, tetapi disimpan di cloud.[2]

3. METODE PENELITIAN

3.1 Prosedur Penelitian

Dalam membuat sistem pendeteksi kebakaran menggunakan mikrokontroler ESP32 diperlukan beberapa prosedur atau tahapan. Adapun tahapan dari penelitian tersebut dapat dilihat pada gambar 3.1 di bawah ini.



Gambar 3.1 Tahap pengerjaan penelitian

Berkaitan dengan gambar di atas, berikut penjelasan dari tahapan pengerjaan penelitian, yaitu:

1. Tahap pengumpulan data dan informasi. Penulis mengumpulkan data serta informasi yang dibutuhkan pada saat membuat system pendeteksi kebakaran. Pada proses ini dilakukan riset yang bertujuan untuk menganalisa hal-hal yang berkaitan dengan teori-teori relevan guna mendukung dalam perancangan dan penyusunan system.

2. Tahap analisis kebutuhan dan perancangan. Penulis menganalisa aspek – aspek apa saja yang dibutuhkan dalam

merancang sistem deteksi kebakaran serta menyusun sebuah desain rangkaian sistem dari sisi *hardware* maupun *software* agar mempermudah dalam perancangan sistem nantinya.

3. Tahap perancangan *software* dan *hardware*. Penulis memulai perancangan alat dengan melakukan pemasangan pada komponen fisik atau *hardware* sebagai instalasi awal, seperti menghubungkan sensor dan piranti lainnya ke mikrokontroler. Selanjutnya, penulis melakukan proses pengkodean program pada *software*.

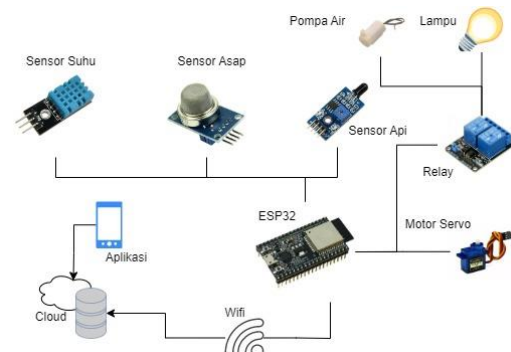
4. Tahap Pengujian Alat. Penulis melaksanakan pengujian pada sistem yang sudah didesain guna memverifikasi terkait fungsi sistem sudah memenuhi kriteria yang ditetapkan atau masih terdapat kekurangan atau kegagalan yang perlu diatasi dengan cepat.

5. Tahap Hasil atau Implementasi Sistem bertujuan untuk memastikan bahwa sistem deteksi kebakaran sudah sesuai dengan harapan dan sanggup beroperasi dengan mulus tanpa ada gangguan sedikitpun.

3.2 Desain Sistem

Sistem yang didesain ini terdiri dari beberapa elemen atau komponen yang akan dipergunakan, antara lain sensor IR Flame, sensor MQ-2, sensor suhu DHT11, ESP32, motor servo dan relay. Sistem ini berfungsi untuk mendeteksi kebakaran dan mengirimkan nilai dari sensor ke arduino. Apabila terdeteksi kebakaran, android akan menerima alarm berupa notifikasi dan pompa air akan diaktifkan untuk mengeluarkan air dan memadamkan api yang di kontrol oleh relay. Nilai dari sistem pendeteksi kebakaran berbasis ESP32 dapat dipantau secara langsung via aplikasi, yang dikirim melalui ESP32 yang terhubung ke jaringan internet dan database. Hal ini memungkinkan pemilik sistem untuk mengambil tindakan dengan cepat dan tepat jika terindikasi tanda-tanda kebakaran.

Sistem ini tak hanya dapat mendeteksi api, melainkan juga dilengkapi dengan dua sensor tambahan yaitu MQ-2 dan DHT11 yang masing-masing sensor berperan guna mengetahui keberadaan asap dan mampu mengukur suhu ruangan.



Gambar 3.2 Desain Alat

3.3 Kebutuhan Sistem

Berdasarkan data yang didapatkan, sistem pendeteksi kebakaran ini memerlukan beberapa kebutuhan guna merealisasikan sistem ini.

3.3.1 Kebutuhan Input

Kebutuhan input atau masukan dari sistem deteksi kebakaran berbasis internet of things dengan perangkat Arduino antara lain sebagai berikut:

1. Data sensor suhu berupa nilai analog
2. Data sensor gas berupa nilai analog
3. Data sensor api berupa nilai digital

3.3.2 Kebutuhan Output

Kebutuhan output atau keluaran dari sistem deteksi kebakaran berbasis internet of things dengan perangkat arduino antara lain sebagai berikut:

1. Data pengukuran sensor suhu yang akan menampilkan nilai suhu dalam satuan derajat celcius
2. Data pengukuran sensor gas yang akan memberikan angka berupa nilai kadar asap
3. Data pengukuran sensor api dengan menampilkan nilai digital 0 dan 1

3.3.3 Kebutuhan Hardware

Kebutuhan hardware atau perangkat keras dari sistem deteksi kebakaran berbasis internet of things dengan perangkat Arduino antara lain sebagai berikut:

1. ESP32 sebagai mikrokontroler dari sistem pendeteksi kebakaran yang akan dirancang
2. Sensor DHT11 yang digunakan untuk mengukur suhu disekitar Sensor MQ-2 yang berfungsi untuk mendeteksi adanya asap
3. Sensor IR Flame yang berguna untuk mendeteksi adanya api.
4. Buzzer digunakan untuk memberikan bunyi alarm saat terdeteksi adanya api, asap dan suhu yang tinggi
5. Kabel jumper digunakan untuk menghubungkan pin – pin dari ESP32 ke pin sensor, relay dan motor servo
6. Motor servo digunakan sebagai penggerak jendela yang akan di kontrol dari aplikasi android
7. Relay 2 channel digunakan untuk memutus dan menyambungkan arus listrik serta menghidupkan dan mematikan pompa air secara otomatis
8. Pompa air digunakan untuk melakukan penyiraman jika terdeteksi adanya api
9. Lampu digunakan sebagai indikator listrik
10. Baterai digunakan untuk sebagai sumber listrik untuk menghidupkan pompa air.

3.3.4 Kebutuhan Software

Kebutuhan software atau perangkat lunak pada sistem deteksi kebakaran berbasis internet of things dengan perangkat Arduino antara lain sebagai berikut:

1. Arduino IDE yaitu software yang digunakan untuk membuat program sistem pendeteksi kebakaran.

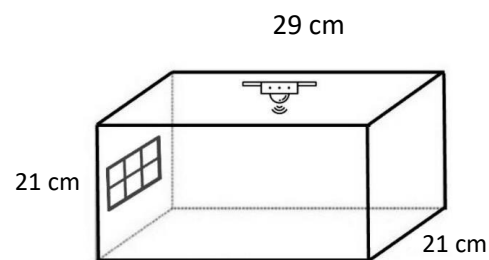
2. Android Studio merupakan perangkat lunak yang dipakai dalam memprogram untuk aplikasi android.

3. Database Firebase digunakan sebagai tempat penyimpanan data dari sensor sekaligus sebagai penghubung antara ESP32 dan aplikasi android.

4. *Smartphone* berfungsi untuk monitoring, mengirimkan notifikasi serta mengontrol sistem yang dibuat dari jarak jauh.

3.4 Perancangan Sistem

Sistem yang dirancang ini akan menampilkan nilai dari data sensor yang telah diletakkan pada suatu ruangan. Dimana data sensor tersebut akan ditampilkan pada aplikasi android. Jadi, sistem ini melakukan pertukaran data antara pembacaan nilai sensor dengan aplikasi android secepat mungkin. Untuk perancangan sistemnya, semua komponen hardware akan dipasang pada sebuah protipe ruangan yang berbentuk kotak tertutup dengan dimensi 29 cm x 21 cm x 21 cm menggunakan bahan dasar akrilik dan kertas karton, dimana sensor nantinya akan diletakkan di bagian atas kotak tersebut, seperti yang terlihat pada gambar 3.3 di bawah.

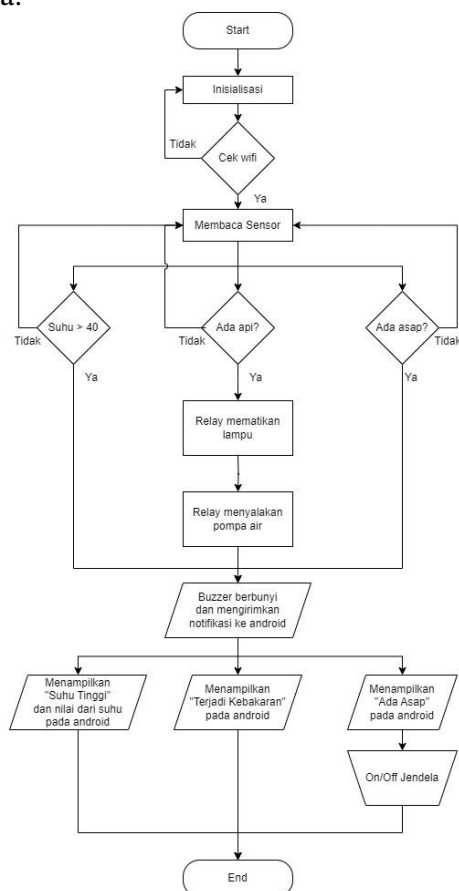


Gambar 3.2 Desain prototipe ruangan

3.4.1 Perancangan Alur Kerja Sistem

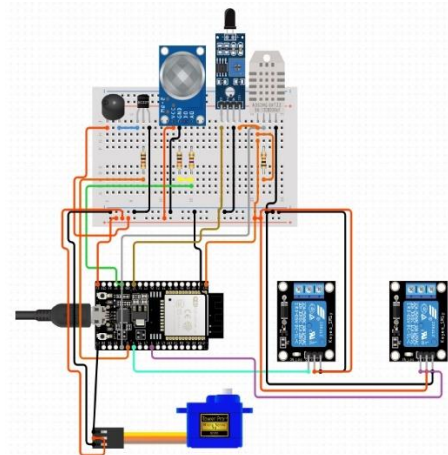
Diagram alir sistem pada gambar 3.4 menjelaskan saat ESP32 dinyalakan, akan menyambungkan ke jaringan wifi, lalu nilai dari sensor – sensor akan diproses oleh ESP32. Jika sensor mendeteksi api, relay akan mematikan listrik dan menyalakan pompa air, setelah itu sistem memberikan

alarm berupa buzzer serta notifikasi ke aplikasi android. Apabila suhu terdeteksi melebihi 40 derajat celcius, aplikasi android akan menerima notifikasi serta buzzer akan berbunyi. Apabila terdeteksi kebakaran asap oleh sensor MQ-2, aplikasi android juga akan menerima notifikasi dan buzzer juga berbunyi. Data dari sensor tersebut kemudian akan dikirimkan ke database server untuk ditampilkan secara *real-time* pada aplikasi android. Lalu saat terdeteksi asap pada aplikasi android, dapat melakukan switch on/off untuk mengontrol jendela.



Gambar 3.3 Diagram Alir Sistem

3.4.2 Perancangan *Hardware*



Gambar 3.4 Skema rangkaian sistem

Pada sistem yang didesain ini, komponen *hardware* atau perangkat keras memainkan peran penting. Komponennya meliputi ESP32, sensor api, DHT11, MQ-2, led, motor servo dan relay. Setiap piranti dihubungkan ke ESP32, dan ESP32 akan melakukan pertukaran data. Skema rangkaian tersebut dapat diamati pada Gambar 3.4 di atas. Terdapat juga sumber daya atau power yang akan mengalir ke semua perangkat.

3.4.3 Perancangan *Software*

Pada sistem pendeteksi kebakaran ini, dibuat sebuah software yang berfungsi untuk menampilkan nilai yang akan dibaca oleh sensor nantinya, serta membuat tombol untuk mengontrol beberapa hardware seperti jendela, listrik dan pompa air. Pada tampilan aplikasinya akan menampilkan nilai dari suhu, keamanan dari ruangan serta keadaan api dan asap pada ruangan. Untuk lebih detailnya mengenai tampilan aplikasi yang akan dibuat dapat dilihat pada gambar 3.6.



Gambar 3.5 Desain aplikasi android

3.5 Proses Pengujian Alat

Pengujian alat untuk sistem ini dilakukan guna menguji fungsi masing-masing sensor serta relay dan servo sehingga dapat bekerja sesuai dengan hasil yang ingin didapatkan dan dapat menampilkan nilai-nilai sensor pada aplikasi android secara *real-time*. Hasil uji coba tersebut akan dicatat ke dalam tabel sebagai data dari hasil uji coba. Jika ingin meninjau hasil dari pengukuran sensor dapat dilihat pada tampilan aplikasi android, karena dengan melihat pada aplikasi nilai pada sensor akan lebih mudah dimengerti. Dengan demikian, nilai pengukuran sensor dapat menjadi acuan untuk mengetahui apakah alat yang digunakan dapat beroperasi sebagaimana mestinya dengan pengujian yang sudah diterapkan.

3.5.1 Pengujian Sensor Api

Dalam melakukan pengujian sensor api, terdapat langkah-langkah yang harus dilakukan yaitu sebagai berikut.

1. Melakukan penyambungan dari ESP32 ke sensor api dengan menyambungkan pin 23 pada ESP32 dihubungkan ke pin D0 sensor api, pin VCC ESP32 ke pin VCC sensor api serta pin GND ESP32 ke pin GND dari sensor api.

2. Melakukan penyambungan dari ESP32 ke buzzer, dengan menyambungkan kabel dari pin 5 ESP32 ke pin (+) buzzer serta pin GND ESP32 ke pin (-) buzzer.

3. Lalu, pada relay 2 channel, untuk pin in1 disambungkan ke pin ESP32 dan pin in2 disambungkan ke pin ESP32, untuk VCC disambungkan ke pin VCC ESP32 serta pin GND ke pin GND ESP32.

4. Lalu menyalakan lilin, kemudian sensor api diletakkan lurus dengan lilin.

5. Selanjutnya meletakkan sensor api dengan jarak yang berbeda-beda dari lilin seperti pada gambar di bawah ini.



Gambar 3.6 Pengujian sensor api pada jarak 70 cm



Gambar 3.7 Pengujian sensor api pada jarak 150 cm



Gambar 3.8 Pengujian sensor api pada jarak 170 cm

6. Mengamati relay (listrik dan pompa air), buzzer dan notifikasi pada android serta keterangan pada aplikasi android.



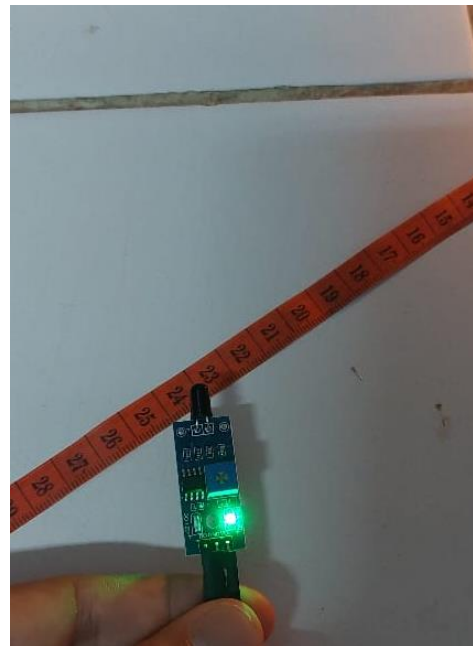
Gambar 3.9 Tampilan terdeteksi kebakaran

7. Kemudian melakukan pengujian dengan mengganti tiap nilai sudut pembacaan sensor ke sumber api, dengan menggunakan tiga sudut perbandingan

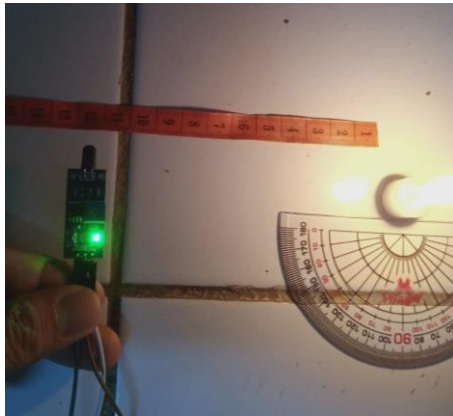
yaitu 30°, 60° dan 90° seperti yang ditampilkan pada gambar – gambar di bawah ini.



Gambar 3.10 Pengujian sensor api dengan sudut 30°



Gambar 3.11 Pengujian sensor api dengan sudut 60°



Gambar 3.12 Pengujian sensor api dengan sudut 90°

3.5.2 Pengujian Sensor Asap

Dalam melakukan pengujian sensor asap, terdapat langkah-langkah yang harus dilakukan yaitu sebagai berikut.

1. Melakukan penyambungan dari ESP32 ke sensor asap dengan menyambungkan pin 34 pada ESP32 dihubungkan ke pin A0 sensor asap, pin VCC ESP32 ke pin VCC sensor asap serta pin GND ESP32 ke pin GND dari sensor asap.
2. Lalu membakar kertas, kemudian sensor asap diletakkan didekat kertas yang dibakar.



Gambar 3.13 Pengujian sensor asap

3. Mengamati buzzer dan notifikasi pada android serta keterangan pada aplikasi android.



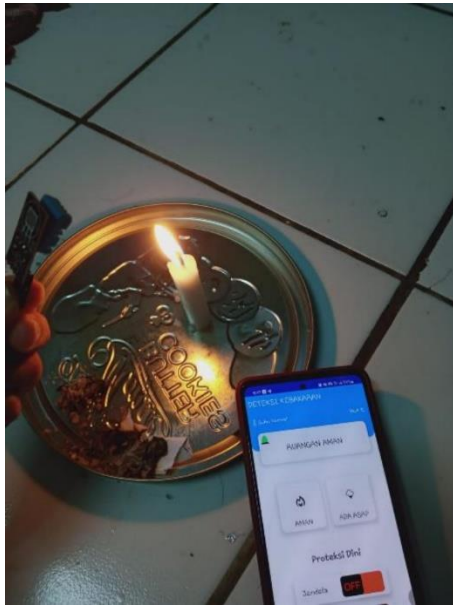
Gambar 3.14 Tampilan terdeteksi asap pada aplikasi

4. Kemudian mengamati nilai kadar asap pada serial monitor.

3.6.1 Pengujian Sensor Suhu

Dalam melakukan pengujian sensor suhu, terdapat langkah-langkah yang harus dilakukan yaitu sebagai berikut.

1. Melakukan penyambungan dari ESP32 ke sensor suhu dengan menyambungkan pin 15 pada ESP32 dihubungkan ke pin out sensor suhu, pin VCC ESP32 ke pin (+) sensor api serta pin GND ESP32 ke pin (-) dari sensor suhu.
2. Kemudian melakukan uji perbandingan pembacaan sensor DHT11 dengan HTC-1 dengan membaca suhu ruangan.
3. Setelah melakukan uji perbandingan, dilakukan pengujian notifikasi dan buzzer untuk sensor DHT11 dengan menyalakan lilin, kemudian sensor suhu diletakkan di atas lilin.



Gambar 3.15 Proses pengujian sensor suhu

4. Mengamati buzzer dan notifikasi pada android
5. Lalu juga mengamati keterangan suhu pada aplikasi android



Gambar 3.16 Tampilan keterangan suhu pada aplikasi

3.6.2 Pengujian Software

Dalam melakukan pengujian software, terdapat langkah-langkah yang harus dilakukan yaitu sebagai berikut.

1. Melakukan pengkodean program sistem untuk ESP32
2. Melakukan upload program ke sistem.
3. Setelah itu, cek serial monitor apakah program sudah berjalan sesuai dengan yang diharapkan.

3.6.6 Pengujian Alat Keseluruhan

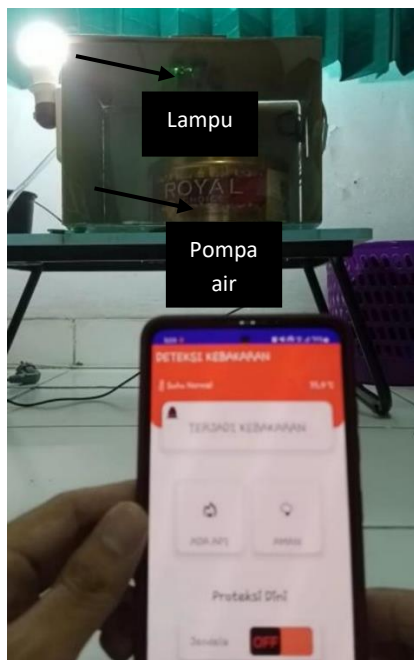
Dalam melakukan pengujian alat secara keseluruhan, dilakukan beberapa tahapan yang dilakukan, yaitu sebagai berikut.

1. Memasang sensor DHT11, sensor MQ-2, sensor api dan motor servo ke bagian dalam prototipe
2. Lalu pada bagian belakang dipasang pompa air yang menggunakan baterai sebagai sumber listrik dan lampu yang keduanya terhubung ke relay.
3. Lalu menghubungkan ESP32 dengan semua komponen *hardware*.
4. Setelah itu, prototipe sistem kebakaran telah selesai dipasang seperti pada gambar 3.20.



Gambar 3.17 Tampilan depan prototipe

5. Setelah melakukan pemasangan ke prototipe selanjutnya dilakukan simulasi kebakaran dengan membakar kertas yang diletakkan di dalam prototipe seperti yang terlihat pada gambar 3.21 di bawah.



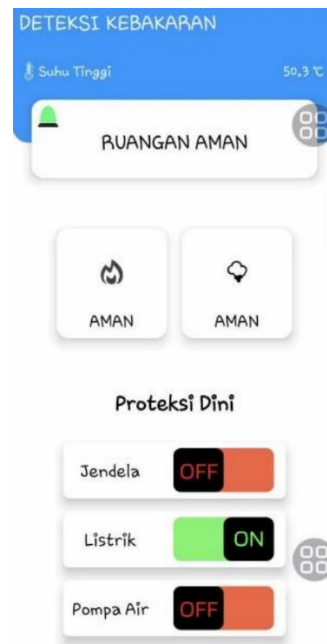
Gambar 3.18 Tampilan awal saat terdeteksi kebakaran

6. Lampu menjadi mati dan pompa air menyiram api, sedangkan pada aplikasi android menampilkan keterangan terjadi kebakaran seperti pada gambar 3.22.



Gambar 3.19 Lampu mati dan pompa air nyala

7. Lalu, suhu menampilkan keterangan “suhu tinggi” pada aplikasi seperti yang terlihat pada gambar 3.23.



Gambar 3.20 Tampilan suhu tinggi

8. Lalu dilakukan pencegahan selanjutnya dengan membuka jendela melalui tombol switch pada aplikasi android seperti yang terlihat pada gambar 3.24.



Gambar 3.21 Jendela Terbuka

4. HASIL DAN PEMBAHASAN

4.1 Hasil Pengujian Sistem

Untuk hasil pengujian sistem yaitu mencakup dari proses pengujian dan pengukuran alat yang dilakukan pada Bab III yang meliputi sensor api, sensor asap, sensor suhu serta program koding sistem.

4.1.1 Hasil Pengujian Sensor Api

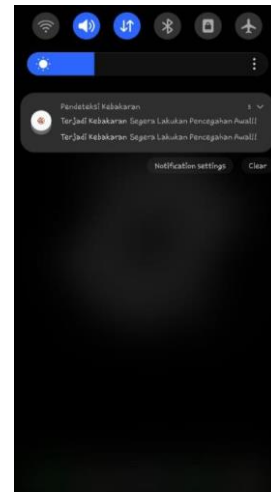
Sensor api adalah jenis sensor yang memiliki tingkat kepekaan dan kecepatan

penerimaan data yang lebih tinggi dibandingkan dengan sensor lainnya. Sensor ini juga memanfaatkan teknologi inframerah untuk mendeteksi keberadaan cahaya api. Sebagai akibatnya, semakin besar sumber api yang ada, jarak deteksinya juga akan semakin jauh. Disini penulis melakukan pengujian simulasi kebakaran dengan menggunakan lilin sebagai sumber api, walaupun pada keadaan di lapangan saat terjadi kebakaran sumber api akan jauh lebih besar. Dilakukan pengujian sensor api didapatkan hasil pengujian yang ditunjukkan pada tabel 4.1.

Tabel 4.1 Hasil pengujian sensor api

Jarak (cm)	Keterangan Aplikasi	Buzzer	Notifikasi	Listrik	Pompa Air
10	Terdeteksi Api	ON	ON	OFF	ON
30	Terdeteksi Api	ON	ON	OFF	ON
40	Terdeteksi Api	ON	ON	OFF	ON
70	Terdeteksi Api	ON	ON	OFF	ON
100	Terdeteksi Api	ON	ON	OFF	ON
120	Terdeteksi Api	ON	ON	OFF	ON
150	Terdeteksi Api	ON	ON	OFF	ON
160	Terdeteksi Api	ON	ON	OFF	ON
170	Aman	OFF	OFF	ON	OFF

Dapat dilihat pada tabel 4.1 saat jarak dari sensor ke titik api sejauh 10 cm – 160 cm, sensor masih dapat mendeteksi adanya api, sehingga buzzer berbunyi dan relay bekerja sesuai yang diharapkan. Android juga menerima notifikasi seperti yang terlihat pada gambar 4.2. Lalu pada jarak lebih dari 170 cm dari titik api, sensor tidak lagi mendeteksi api, dan buzzer menjadi mati. Sehingga dapat disimpulkan dari proses pengujian sensor api dan juga dari notifikasi serta tampilan android dapat berjalan lancar tanpa adanya kendala serta bekerja sesuai yang diharapkan.



Gambar 4.1 Tampilan notifikasi terjadi kebakaran pada android

Lalu melakukan pengujian sensor api dengan membandingkan tiap sudut ke titik sumber api seperti yang ditunjukkan pada tabel di bawah.

Tabel 4.2 Hasil pengujian sensor api dengan perbandingan sudut

No.	Jarak (cm)	Sudut dari sensor ke titik api		
		30°	60°	90°
1	6	ON	ON	ON
2	12	ON	ON	OFF
3	16	ON	ON	OFF
4	19	ON	ON	OFF
5	23	ON	OFF	OFF
6	28	ON	OFF	OFF
7	33	OFF	OFF	OFF
8	42	OFF	OFF	OFF

Berdasarkan tabel 4.2 hasil pengujian sensor api dapat dilihat pada sudut 30° sensor tidak dapat mendeteksi api lagi pada jarak 33 cm, lalu pada sudut 60° tidak dapat mendeteksi api saat jarak 23 cm, kemudian jarak 12 cm pada sudut 90°. Jadi, dapat disimpulkan bahwa detektor dari sensor api dapat mendeteksi api jika sudut sumber titik apinya semakin kecil dan sebaliknya tidak dapat mendeteksi api saat sudut sumber titik apinya semakin besar.

4.1.2 Hasil Pengujian Sensor Asap

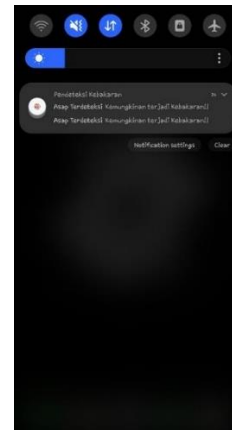
Untuk pengujian sensor asap disini menggunakan sensor MQ-2. Dimana sensor ini dapat mendeteksi asap yang didalamnya terkandung gas monoksida (CO). Jadi, sensor ini sangat cocok untuk mendeteksi adanya asap yang disebabkan oleh kebakaran. Namun, sensor ini sangat berbeda dari sensor api yang mempunyai sensitifitas yang cepat, sensor ini kurang sensitif dan harus menunggu beberapa detik agar asap masuk ke dalam tabung sensor sehingga dapat mendeteksi adanya asap.

Berdasarkan pengujian alat yang dilakukan pada sensor asap, didapatkan hasil dengan menampilkan kadar CO yang terkandung pada asap seperti pada tabel 4.3.

Tabel 4.3 Hasil pengujian sensor asap

Kadar Asap (PPM)	Keterangan Aplikasi	Notifikasi	Buzzer
235	Aman	OFF	OFF
246	Aman	OFF	OFF
258	Aman	OFF	OFF
271	Aman	OFF	OFF
292	Aman	OFF	OFF
304	Aman	OFF	OFF
347	Aman	OFF	OFF
378	Aman	OFF	OFF
402	Ada Asap	ON	ON
413	Ada Asap	ON	ON
432	Ada Asap	ON	ON
446	Ada Asap	ON	ON
458	Ada Asap	ON	ON

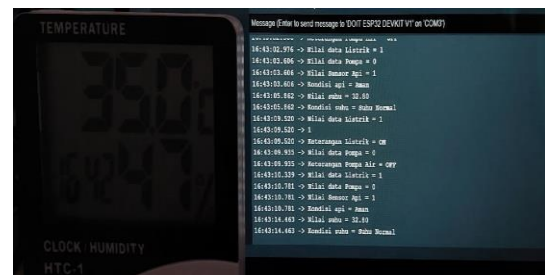
Dapat dilihat pada tabel 4.3 saat kadar asap bernilai 235 – 378 PPM keterangan pada aplikasi menunjukkan “Aman”, serta notifikasi android dan buzzer tidak aktif saat kadar asap bernilai 402 - 458 aplikasi android memberikan notifikasi seperti yang terlihat pada gambar 4.2 serta buzzer berbunyi dan pada aplikasi memberikan keterangan “Ada Asap”. Sehingga dapat disimpulkan, bahwa sensor suhu telah bekerja sesuai dengan yang diharapkan.



Gambar 4.2 Tampilan notifikasi asap terdeteksi pada android

4.1.3 Hasil Pengujian Sensor Suhu

Dalam pengujian sensor suhu menggunakan sensor berjenis DHT11, dimana sensor ini mendeteksi suhu ruangan. Sebelumnya dilakukan uji pembacaan sensor DHT11 dengan membandingkan pembacaan suhu dari alat HTC yang terlihat seperti pada gambar 4.3 dibawah.



Gambar 4.2 Uji perbandingan sensor DHT11 dengan HTC-1

Lalu, didapatkan hasil pengujian perbandingan antara sensor DHT11 dan HTC-1 yang ditunjukkan pada tabel 4.4 dibawah.

Tabel 4.4 Hasil pengujian perbandingan DHT11 dan HTC-1

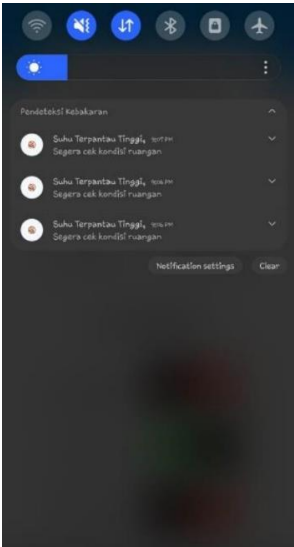
No.	DHT11 (°C)	HTC (°C)	Selisih Pengukuran	Nilai Error (%)
1	33.3	34.7	1.4	4.03
2	32.8	34.8	2.0	5.74
3	32.8	35.0	2.2	6.28
4	33.3	35.3	2.0	5.66
5	33.3	35.7	2.4	6.72
6	33.3	36.0	2.7	7.5
Rata – Rata Error (%)				5.98

Berdasarkan hasil tabel 4.4 terlihat beberapa perbedaan antara pembacaan sensor DHT11 dengan HTC, Sensor DHT11 menampilkan data yang lebih akurat dibandingkan dengan HTC-1. Hal tersebut dikarenakan sensor DHT11 yang dapat dilihat update setiap 1 detik melalui serial monitor, berbeda dengan HTC-1 yang perubahannya hanya dapat dilihat apabila terjadi kenaikan ataupun penurunan suhu. Selanjutnya dilakukan pengujian buzzer dan notifikasi android yang disini menggunakan api lilin sebagai sumber panasnya, dan terlihat perubahan suhu pada hasil pembacaan sensor seperti yang ditunjukkan pada tabel 4.5.

Tabel 4.5 Hasil pengujian sensor suhu

Nilai Suhu	Keterangan Aplikasi	Notifikasi	Buzzer
33.0 °C	Suhu Normal	OFF	OFF
35.4 °C	Suhu Normal	OFF	OFF
36.9 °C	Suhu Normal	OFF	OFF
37.8 °C	Suhu Normal	OFF	OFF
38.6 °C	Suhu Normal	OFF	OFF
39.5 °C	Suhu Normal	OFF	OFF
41.8 °C	Suhu Tinggi	ON	ON
43.4 °C	Suhu Tinggi	ON	ON

Pada tabel 4.5 menunjukkan suhu 33 °C - 39.5 °C, aplikasi memberikan keterangan suhu normal, notifikasi dan buzzer tidak aktif. Pada saat nilai suhu menunjukkan nilai 41.8 °C – 43.4 °C, aplikasi android menerima notifikasi seperti pada gambar 4.4 serta buzzer berbunyi. Sehingga dapat disimpulkan bahwa sensor suhu bekerja tanpa kendala dan sesuai yang diharapkan.

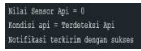
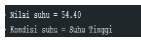


Gambar 4.3 Tampilan notifikasi suhu tinggi pada android

4.1.4 Hasil Pengujian Software

Untuk pengujian software atau perangkat lunak, dilakukan pengujian terhadap program koding yang dilakukan pada mikrokontroler ESP32 terhadap sensor – sensor yang digunakan. Dengan menampilkan kode program yang digunakan beserta hasil yang didapatkan setelah melakukan eksekusi program. Untuk hasil pengujiannya dapat dilihat pada tabel 4.6 di bawah.

Tabel 4.6 Hasil pengujian program software

Program	Eksekusi	Kondisi
		Baik
		Baik
		Baik

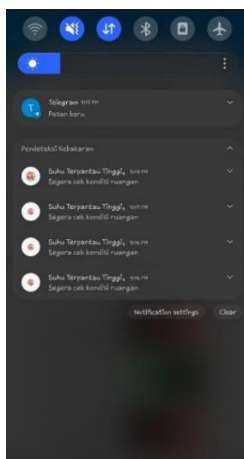
4.1.5 Hasil Pengujian Sistem Keseluruhan

Dalam melakukan pengujian sistem secara keseluruhan, dilakukan pengujian dengan cara membakar kertas yang berada di dalam prototipe, dan didapatkan hasil pengujian secara keseluruhan pada tabel 4.7 di bawah.

Tabel 4.7 Hasil pengujian secara keseluruhan

Sensor Api	Sistem Proteksi					
	Listrik		Pompa Air		Buzzer	
	Status	Delay (s)	Status	Delay (s)	Status	Delay (s)
ON	OFF	5	ON	5	ON	6
ON	OFF	6	ON	6	ON	7
ON	OFF	5	ON	5	ON	6
ON	OFF	5	ON	5	ON	6
ON	OFF	6	ON	6	ON	7

Pada tabel 4.7 sensor api memberikan indikator ON, sehingga aplikasi memberikan notifikasi pada android dan saat sensor api memberikan indikator OFF, android tidak memberikan notifikasi. Saat pengukuran sensor asap, ketika nilai kadar asap 198 PPM – 357 PPM, android tidak memberikan notifikasi dan saat kadar asap melebihi nilai 402, android menerima notifikasi. Disitu juga terlihat, ketika nilai suhu menunjukkan di atas 41.3 °C, aplikasi android menerima notifikasi seperti pada gambar 4.5. Sehingga dapat disimpulkan bahwa semua sensor dan notifikasi dapat bekerja dalam satu sistem tanpa kendala dan sesuai yang diharapkan.



Gambar 4.4 Notifikasi suhu terpantau cukup tinggi

Kemudian terjadi juga proteksi dini yang akan dilakukan sistem saat terjadinya kebakaran. Dengan memverifikasi berapa lama kinerja relay dan buzzer terhadap hasil pembacaan dari sensor api seperti yang ditampilkan pada tabel 4.8 di bawah.

Tabel 4.8 Hasil pengujian waktu sistem melakukan proteksi dini

No	Sensor Api	Sensor Asap (PPM)	Sensor Suhu (°C)	Notifikasi		
				Api	Asap	Suhu
1	ON	198	32	ON	OFF	OFF
2	ON	203	35.9	ON	OFF	OFF
3	ON	226	41.3	ON	OFF	ON
4	OFF	263	45.7	OFF	OFF	ON
5	OFF	301	47.9	OFF	OFF	ON
6	OFF	357	51.5	OFF	OFF	ON
7	OFF	402	48.8	OFF	ON	ON
8	OFF	475	46.4	OFF	ON	ON
9	OFF	498	46.2	OFF	ON	ON
10	OFF	482	45	OFF	ON	ON
11	OFF	478	44.7	OFF	ON	ON
12	OFF	465	42.1	OFF	ON	ON

Dapat dilihat pada tabel di atas, saat sensor api mendeteksi adanya api, sistem proteksi membutuhkan waktu sekitar 5 – 6 detik untuk mematikan listrik dan menyalakan pompa air, sedangkan buzzer akan berbunyi pada waktu 6 – 7 detik. Sehingga kinerja sistem proteksi yang dilakukan telah sesuai yang diharapkan.

5. KESIMPULAN

Berdasarkan pengujian yang telah dilakukan menggunakan ESP32, sensor suhu DHT11, sensor gas MQ-2, sensor api, motor servo dan relay dapat diperoleh beberapa kesimpulan sebagai berikut:

1. Ketiga sensor mampu mendeteksi adanya api, asap dan juga suhu ruangan. Sensor api dapat mendeteksi api hingga 160 cm pada sudut 0° dan saat jarak 170 cm tidak dapat lagi mendeteksi adanya api dan pada jarak 12 cm di sudut 90°. Sehingga semakin besar sudut antara detektor sensor api dengan sumber api, maka pembacaan sensor menjadi berkurang. Sensor asap mendeteksi kadar asap saat bernilai lebih dari 402 PPM dan untuk sensor suhu mempunyai tingkat keakuratan yang baik karena dapat melakukan update secara cepat dibandingkan dengan alat ukur suhu HTC-1 serta saat suhu menunjukkan nilai

lebih dari 41.8 °C akan memberikan keterangan suhu tinggi.

2. Android memberikan notifikasi serta buzzer akan berbunyi ketika ketiga sensor mendeteksi terjadinya kebakaran.

3. Sistem proteksi juga bekerja dengan memberikan proteksi berupa listrik mati dan pompa air menyala untuk mematikan api. Sistem proteksi tersebut mempunyai delay dalam selang waktu 5-6 detik saat sensor api mendeteksi adanya api.

Sehingga dapat disimpulkan bahwa sistem ini dapat bekerja tanpa kendala dan sesuai dengan apa yang diharapkan serta dapat mengurangi risiko kerusakan yang disebabkan oleh kebakaran.

5.1 Saran

Berdasarkan hasil penelitian yang telah dilakukan, ada beberapa saran yang bisa menjadi pedoman untuk mengembangkan sistem ini.

1. Untuk penelitian selanjutnya, bisa menambahkan sistem yang dapat mendeteksi penyebab kebakaran tersebut salah satunya dapat mendeteksi kebocoran gas LPG.

2. Untuk bagian proteksi dini, bisa menambahkan perintah agar lampu dan pompa bisa di kontrol secara IoT bukan hanya motor servo saja.

3. Untuk tampilan aplikasi dapat dibuat lebih kreatif dan informatif lagi.

4. Menambahkan Exhaust Fan untuk mendinginkan ruangan.

DAFTAR PUSTAKA

- [1] Arifin, Jauhari, Leni, N. Z., Hermawansyah. (2016). Perancangan Murottal Otomatis Menggunakan Mikrokontroler Arduino Mega2560. *Jurnal Media Infotama*, 12(1), 89 – 98.
- [2] Sanadi, E.A.W., Andani, A., & Dewiani. (2018). Pemanfaatan Realtime Database di Platform Firebase Pada Aplikasi E-Tourism Kabupaten Nabire. *JPE-UNHAS*, 22(1), 20-26.
- [3] Wardoyo, S., Jajang, S., & Anggoro, S. P. (2013). Rancang Bangun Alat Uji Karakteristik Motor DC Servo, Battery, dan Regulator untuk Aplikasi Robot Berkaki. *SETRUM*, 2(2), 54-59.
- [4] Efrianto, Ridwan, & Iman, F. (2016). Sistem Pengaman Motor Menggunakan Smartcard Politeknik Negeri Batam. *Jurnal Integrasi*, 8(1), 1-5.
- [5] Mus Mulyadi Usman, “Rancang Bangun Aplikasi Monitoring Ketinggian Air Sungai Berbasis Internet of Things Menggunakan Amazon Web Service,” *J. Tek. Elektro dan Komput.*, vol. 9, no. 2, pp. 73–80, 2020
- [6] K. L. Yana, K. R. Dantes, and N. A. Wigrha, “Rancang Bangun Mesin Pompa Air Dengan Sistem Recharging,” *J. Pendidik. Tek. Mesin Undiksha*, vol. 5, no. 2, 2017,
- [7] N. H. L. Dewi, M. F. Rohmah, and S. Zahara, “Prototype Smart Home Dengan Modul Nodemcu Esp8266 Berbasis Internet of Things (Iot),” *J. Tek. Inform.*, p. 3, 2019.
- [8] Sulistyowati, R., Sujono, H. A., & Musthofa, A. K. (2015). Sistem Pendeteksi Banjir berbasis Sensor Ultrasonik dan Mikrokontroler dengan Media Komunikasi Gateway.
- [9] Aldi Razor,. “Module Relay Arduino Pengertian, Gambar Dan Skema,” ; (<https://www.aldyrazor.com/2020/05/modul-relay-arduino.html>) 2020- 5