**PERANCANGAN SISTEM KEAMANAN KANTOR BERBASIS IOT DENGAN MENGGUNAKAN MIKROKONTROLER ESP32**

**SKRIPSI**

**Oleh:**

**VINCENT**

**NIM. 2044135**



**JENJANG PENDIDIKAN STRATA-1**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**STMIK TIME**

**MEDAN**

**2024**

**LEMBAR PENGESAHAN**

**PERANCANGAN SISTEM KEAMANAN KOMPUTER BERBASIS IOT DENGAN MENGGUNAKAN MIKROKONTROLER ESP32**

**SKRIPSI**

Diajukan Untuk Melengkapi Persyaratan Guna

Mendapat Gelar Sarjana Strata Satu

Program Studi Teknik Informatika

Oleh :

**VINCENT**

**NIM. 2044135**

Disetujui Oleh:

Dosen Pembimbing I Dosen Pembimbing II

(Octara Pribadi, M.Kom) (Hendri, M.Kom)

Medan, Juni 2024

Diketahui dan Disahkan Oleh :

Ketua STMIK TIME Ketua Program Studi

( Dr. Edi Wijaya, S.Kom,M.Kom, MM) (Robet,M.Kom)

**ABSTRAK**

Kemajuan teknologi *Internet of Things* (IoT) telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk di bidang keamanan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem keamanan kantor berbasis IoT menggunakan mikrokontroler ESP32. Sistem ini dirancang untuk meningkatkan keamanan dengan mengintegrasikan berbagai sensor dan aktuator yang terhubung melalui jaringan internet. Mikrokontroler ESP32 dipilih sebagai inti dari sistem karena keunggulannya dalam hal konektivitas Wi-Fi, efisiensi energi, dan kemampuan pemrosesan data. Sistem ini mencakup berbagai komponen seperti kamera ESP32-CAM untuk pengawasan visual, sensor gerak PIR untuk deteksi aktivitas, dan modul relay untuk kontrol akses. Data yang diperoleh dari sensor akan diproses oleh ESP32 dan dikirimkan ke server Firebase untuk penyimpanan dan analisis lebih lanjut. Sistem juga dilengkapi dengan integrasi bot Telegram untuk memberikan notifikasi real-time kepada pengguna ketika terdeteksi aktivitas mencurigakan. Pengguna dapat berinteraksi dengan sistem melalui aplikasi Telegram untuk menerima foto atau video dari kamera, serta untuk mengendalikan perangkat secara jarak jauh. Hasil dari implementasi ini menunjukkan bahwa sistem mampu memberikan respons cepat terhadap kejadian yang terdeteksi dan memberikan notifikasi secara real-time. Sistem ini juga menawarkan fleksibilitas dan skalabilitas tinggi yang memungkinkan penambahan sensor atau aktuator di masa depan sesuai kebutuhan. Dengan demikian, sistem keamanan kantor berbasis IoT ini diharapkan dapat meningkatkan keamanan secara signifikan dan memberikan rasa aman bagi pengguna.

**Kata kunci: *Internet of Things, IoT*, ESP32, keamanan kantor, sensor, Firebase, Telegram**

***ABSTRACT***

*The advancement of Internet of Things (IoT) technology has brought significant changes in various aspects of life, including security. This research aims to design and implement an office security system based on IoT using the ESP32 microcontroller. The system is designed to enhance security by integrating various sensors and actuators connected via the internet. The ESP32 microcontroller was chosen as the core of the system due to its advantages in Wi-Fi connectivity, energy efficiency, and data processing capabilities. The system includes various components such as the ESP32-CAM camera for visual surveillance, PIR motion sensors for activity detection, and relay modules for access control. Data obtained from the sensors will be processed by the ESP32 and sent to the Firebase server for storage and further analysis. The system is also equipped with Telegram bot integration to provide real-time notifications to users when suspicious activity is detected. Users can interact with the system through the Telegram application to receive photos or videos from the camera and to remotely control devices. The results of this implementation show that the system is capable of providing quick responses to detected incidents and delivering real-time notifications. The system also offers high flexibility and scalability, allowing for the addition of sensors or actuators in the future as needed. Thus, this IoT-based office security system is expected to significantly enhance security and provide a sense of safety for users.*

***Keywords: Internet of Things, IoT, ESP32, office security, sensors, Firebase, Telegram***

**KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan karunia-Nya penulis dapat meyelesaikan skripsi ini dengan baik. Adapun tujuan dari penulisan skripsi ini adalah untuk melengkapi persyaratan guna mendapatkan gelar Sarjana Strata Satu Program Studi Teknik Informasi serta untuk menambah wawasan penulis dan pembaca skripsi yang berjudul “Perancangan Sistem Keamanan Kantor Berbasis IOT Dengan Menggunakan ESP32.”

Dalam proses penyusunan skripsi ini, banyak terdapat kendala dan kesulitan yang dihadapi oleh penulis, namun berkat bantuan, dorongan dan nasehat dari berbagai pihak, terutama bimbingan dan pengarahan dari dosen pembimbing kepada penulis hingga skripsi ini dapat diselesaikan dengan baik, maka melalui kesempatan ini, penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Octara Pribadi, M.Kom, selaku Dosen Pembimbing I yang telah memberikan arahan dan bimbingan dalam menyelesaikan penelitian ini.

2. Bapak Hendri, M.Kom, selaku Dosen Pembimbing II yang juga telah membantu untuk membimbing dan mengarahkan dalam menyelesaikan penelitian ini .

3.Bapak Simon Kangga Lee, selaku Ketua Yayasan STMIK TIME Medan.

4. Bapak Dr. Edi Wijaya, S.Kom,M.Kom, MM, selaku Ketua STMIK TIME Medan.

5. Seluruh Dosen STMIK TIME Medan, yang telah banyak memberikan ilmu pengetahuan kepada penulis selama perkuliahan.

6. Orang tua serta rekan-rekan penulis yang telah memberikan motivasi dan bantuan dalam menyelesaikan skripsi ini.

Penulis menyadari bahwa masih terdapat banyak kekurangan dalam penulisan skripsi ini. Oleh karena itu, penulisakan dengan senang hati menerima kritik dan saran yang bersifat membangun dari semua pihak. Akhir kata, penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak. Terima kasih.

Medan, Juni 2024

Penulis

Vincent

**DAFTAR ISI**

**DAFTAR ISI**  i

**BAB I PENDAHULUAN** **1**

* 1. Latar Belakang Masalah 1
  2. Rumusan Masalah 3
  3. Batasan Masalah 4
  4. Tujuan dan Manfaat Penelitian 4
  5. Metodologi Penelitian 6

1.5.1. Jadwal Penelitian 6

1.5.2. Bahan dan Alat 6

1.5.3. Metode Pengumpulan Data 7

1.6. Analisis Sistem 7

1.7 Perancangan Sistem 8

1.8 Sistematika Penulisan 9

**BAB II TINJAUAN PUSTAKA**  **11**

2.1 Internet 11

2.2 Internet of Things 13

2.3 Sistem Keamanan 14

2.4 MikroKontroler ESP32-CAM 15

2.5 Sensor Pir 17

2.6 Arduino IDE 19

2.7 Penelitian Terdahulu 21

**BAB III METODE PENELITIAN 26**

3.1 Analisis 26

3.2 Perancangan 35

**BAB IV HASIL DAN PEMBAHASAN 39**

4.1 Hasil 39

4.2 Pembahasan 42

4.2.1 Kelebihan Sistem Usulan 44

4.2.2 Kelemahan Sistem Usulan 46

4.2.3 Penjelasan Pergantian Blynk 47

**BAB V KESIMPULAN DAN SARAN 49**

5.1 Kesimpulan 49

5.2 Saran 50

**DAFTAR PUSTAKA 52**

**DAFTAR TABEL**

Tabel 1.1. Jadwal Penelitian 5

Tabel 4.1 Pengujian Sensor PIR 42

**DAFTAR GAMBAR**

Gambar 2.1. Skematik Mikrokontroller ESP32 17

Gambar 2.2. Skematik Sensor PIR HC-SR501 18

Gambar 2.3. *Interface Arduino IDE 2.2.1* 20

Gambar 3.1. Diagram Arsitektur Sistem 26

Gambar 3.2. Skematik Sensor PIR dan Mikrokontroler ESP32 29

Gambar 3.3. *Flowchart* Metode 34

Gambar 4.1 Rangkaian PIR-ESP32 39

Gambar 4.2 Halaman Utama 40

Gambar 4.3 Halaman *Log* Gambar 41

**BAB I**

**PENDAHULUAN**

## Latar Belakang Masalah

*Internet of Things* (IoT) merupakan sebuah teknologi yang memungkinkan segala jenis perangkat, seperti komputer, telepon pintar, tablet, televisi pintar, dan perangkat rumah yang dilengkapi dengan sensor, aktuator, dan perangkat lunak, untuk terhubung satu sama lain melalui infrastruktur jaringan seperti Internet. Dengan adanya konektivitas ini, perangkat-perangkat tersebut dapat saling berkomunikasi dan bertukar data. Hal ini menciptakan bentuk baru komunikasi antara manusia dengan manusia, manusia dengan perangkat, dan juga antara perangkat dengan perangkat[1].

Sistem keamanan kantor adalah sebuah sistem yang mengendalikan dan mengawasi keamanan suatu kantor yang dimana memiliki sensor, *actuator*, *display* dan elemen komputer di dalamnya[2]. Dalam sistem keamanan diterapkan untuk menjaga keamanan suatu kantor dari setiap aktivitas manusia yang tidak diinginkan misalnya adanya aktivitas manusia yang dilakukan bukan pada waktunya. Kemudian sistem ini akan memberikan pemberitahuan atau notifikasi secara *real time* kepada pengguna ketika ada kejadian atau aktivitas dalam sebuah ruangan pada waktu tertentu.

Pada saat ini banyak pelaku kejahatan melakukan segala cara untuk memenuhi kebutuhan hidupnya. Menurut Badan Pusat Statistik (BPS), kejahatan pada lingkungan kantor selalu meningkat setiap tahunnya[3]. Kejahatan yang kerap terjadi pada wilayah perkantoran merupakan pencurian, banyak tindakan yang telah dilakukan untuk meningkatkan keamanan pada lingkungan kantor seperti ruangan yang menyimpan berkas penting dan hanya karyawan tertentu saja yang boleh masuk.Walaupun dalam kantor ada CCTV tetapi pada waktu malam pengawas CCTV ataupun pihak kemanan mungkin tidak akan terlalu memperhatikan dengan teliti karena adanya faktor seperti tingkat kewaspadaan rendah .Oleh karena itu perlu adanya peningkatan pada sistem keamanan kantor berbasis *Internet of Things* (IoT) yang mengirimkan pemberitahuan kepada pihak keamanan apabila ada pergerakan. Aktivitas di malam hari karena tidak sesuai dengan pola perilaku yang normal atau diharapkan dalam suatu konteks tertentu. Aktivitas di malam hari sering kali dianggap sebagai indikasi adanya potensi ancaman, kejahatan, atau tindakan illegal.Menggunakan mikrokontroler ESP32 sebagai alat untuk memantau.ruangan. Dengan adanya teknologi ini diharapkan berkurangnya tindak kriminal pada lingkungan kantor[4]. IoT sendiri merupakan suatu konsep dimana objek tertentu mempunyai kemampuan untuk berkomunikasi lewat jaringan tanpa memerlukan adanya interaksi dari manusia.

Berdasarkan pada penelitian sebelumnya yang dilakukan oleh M.Reza Ferdiansyah,Arief Wibowo, Subandi,Wahyu Pramusinto dalam jurnal yang berjudul “Perancangan Robot Monitoring Keamanan Kantor Menggunakan Mikrokontroler Esp32CAM Berbasis *Internet Of Things*”. Mereka merancang sebuah robot mikrokontroler yang dirancang untuk mendeteksi suara kemudian merekam gambar yang ada di depannya dan datanya tersimpan dalam sebuah web browser sebagai *log*[5].

Dari permasalahan tersebut, maka dibahas mengenai kendala lalu dipecahkan dengan menggunakan *Internet Of Things* (IoT). Kontrol sistem keamanan dapat langsung dilakukan dengan smartphone android yang dimiliki.Berbeda dengan CCTV sekarang yang hanya dapat merekam jika ada pergerakan dan tidak dapat melaporkan pergerakan tersebut. Dengan sistem keamanan kantor maka sistem pengamanan yang terintegrasi melalui jaringan *Wireless Fidelity* dapat dikendalikan dengan jarak jauh menggunakan perangkat android dan suatu ruangan kantor yang dilengkapi mikrokontroler ESP32 yang juga dilengkapi dengan sensor infrared untuk mendeteksi pergerakan untuk mengaktifkan sensor kamera[6] .Berdasarkan latar belakang di atas, peneliti tertarik untuk melakukan penelitian mengenai Sistem Keamanan Kantor dengan mengambil judul **“PERANCANGAN SISTEM KEAMANAN KANTOR BERBASIS IOT DENGAN MENGGUNAKAN MIKROKONTROLER ESP32”**

## Rumusan Masalah

Berdasarkan latar belakang masalah tersebut maka perumusan masalah adalah sebagai berikut:

1. Bagaimana merancang sistem keamanan kantor agar dapat mendeteksi adanya pergerakan pada waktu malam?
2. Bagaimana sistem ini memberikan notifikasi atau pemberitahuan kepada pemilik kantor atau pihak yang berwenang jika terdeteksi pergerakan?
3. Bagaimana metode atau algoritma yang akan dipakai pada sistem keamanan kantor tersebut?

## Batasan Masalah

Agar penelitian ini terarah dan tidak menyimpang dari permasalahan yang telah dirumuskan, Sistem keamanan kantor yang dibuat hanya sebatas pemikiran penulis yaitu

* 1. Sistem keamanan yang disediakan hanya sebatas memakai Mikrokontroler Esp32 Cam, dan sensor PIR Motion Detector 12V
  2. Sensor yang dipakai hanya akan disambungkan pada perangkat yang tersedia
  3. Sistem monitoring yang dibuat hanya akan sebatas untuk mendeteksi dan melaporkan adanya pergerakan
  4. Sistem yang dibuat hanya akan diterapkan pada satu ruangan dan data yang terekam hanya akan dibuat sebagai log
  5. Sistem yang dibuat akan dibangun dalam sebuah aplikasi android
  6. Sistem yang dibuat di utamakan mendeteksi pergerakan di waktu yang ditentukan
  7. Sensor PIR yang dipakai hanya mendeteksi maksimal 7 meter dan sudut deteksinya hanya 120 derajat

## Tujuan dan Manfaat Penelitian

Tujuan penelitian ini adalah

1. Memanfaatkan sistem keamanan kantor agar dapat mendeteksi dan melaporkan aktivitas kepada tim keamanan
2. Menguji sensor yang digunakan diharapkan dapat membantu meningkatkan akurasi sensor-sensor lain
3. Mempermudah pihak yang berwenang atau tim keamanan jika ada kejadian dengan melihat data yang tersimpan

Adapun manfaat yang dapat diperoleh dari penelitian ini adalah:

1. Manfaat Teoritis
2. penelitian ini dapat berkontribusi pada penemuan dan inovasi baru serta meningkatkan pemahaman tentang keamanan IoT secara keseluruhan.
3. Penelitian ini dapat memberikan landasan untuk memilih sensor dan teknologi yang sesuai untuk tujuan keamanan tertentu.
4. Manfaat Praktis

Secara praktis penelitian ini dapat bermanfaat sebagai berikut:

1. Bagi tim keamanan:

Membantu meningkatkan pemantauan, deteksi dini, dan respon terhadap situasi keamanan

1. Bagi Karyawan kantor

dapat memberikan rasa aman dan kenyamanan bagi karyawan yang bekerja dalam kantor tersebut

## Metodologi Penelitian

Penelitian ini dimulai dari November 2023 dan berakhir pada April 2024, Penelitian di lapangan bertujuan untuk mengumpulkan data yang diperlukan dalam proses perancangan dan pembuatan sistem keamanan kantor . Berikut ini ditampilkan terkait jadwal penelitian dan pembuatan sistem yang akan dilakukan pada penelitian ini antara lain sebagai berikut

1. **Jadwal Penelitian**

**Tabel 1.1 Tabel Jadwal Penelitian**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Waktu  Kegiatan | 2023 | | | | | | | | 2024 | | | | | | | | | | | | | | | | |
| November | | | | Desember | | | | Januari | | | | Februari | | | | Maret | | | | April | | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Perumusan Masalah |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengumpulan Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analisis Sistem Keamanan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Perancangan Sistem Keamanan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Uji Coba Sistem Keamanan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penulisan Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **Bahan dan Alat**

Adapun bahan dan alat yang digunakan yaitu berupa *hardware* dan *software* yang digunakan dalam proses penelitian.

*Hardware* yang digunakan yaitu laptop dengan spesifikasi:

1. Processor Intel core i3-10300F
2. RAM 12GB
3. Mikrokontroler ESP32 CAM
4. Sensor PIR HC-SR501

*Software* yang digunakan, yaitu: Arduino IDE 2.1.0

1. **Metode Pengumpulan Data**

Dalam mengumpulkan data yang diperlukan untuk merancang sistem keamanan kantor, penulis menggunakan Metode Studi Pustaka Untuk mendapatkan data yang lebih akurat dan objektif, penulis melakukan studi pustaka dengan membaca referensi berupa buku dan jurnal penelitian serta mencari informasi dari internet dengan tujuan mendapatkan data yang memiliki hubungan atau kaitan dengan topik penelitian.

## Analisis Sistem

Penulis menganalisis terlebih dahulu dengan mengevaluasi data, informasi dan cara perancangan sedemikian rupa sehingga menghasilkan gambaran awal yang akan dikembangkan lebih lanjut. Dengan data yang telah tersedia, maka dibuatlah suatu sistem keamanan yang diharapkan dapat menjaga keamanan sesuai dengan yang diharapkan.

## Perancangan Sistem

Dalam perancangan sistem keamanan penulis akan menganalisis kebutuhan Fungsional dan Non Fungsional. Kebutuhan Fungsional adalah kebutuhan yang berisi proses-proses apa saja / layanan apa saja yang nantinya harus disediakan oleh sistem, mencakup bagaimana sistem harus bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu. Kebutuhan Non Fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. kebutuhan fungsional juga sering disebut sebagai batasan layanan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses, standarisasi dan lain lain[7].

Sebelum melakukan tahap pembangunan keamanan, penulis akan melakukan perancangan terlebih dahulu terhadap tampilan atau *user interface* dari sistem keamanan yang akan dibangun serta merancang beberapa skenario sistem keamanan yang akan dijadikan sebagai pendukung dalam membangun sistem keamanan. Dalam merancang sistem keamanan tersebut, penulis melakukan analisis terhadap data yang sudah ada, kemudian melakukan penyusunan materi tersebut untuk dapat dirancang. Setelah terbentuk, maka dilakukan perancangan sistem hingga siap untuk dapat diuji aplikasinya.

## Sistematika Penulisan

Skripsi ini disusun dengan sistematika sebagai berikut:

**BAB I PENDAHULUAN**

Dalam bab ini, penulis membahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, analisis sistem serta perancangan sistem.

**BAB II TINJAUAN PUSTAKA**

Dalam bab ini, penulis memaparkan dan menguraikan teori-teori yang berhubungan dengan judul penulisan skripsi.

**BAB III METODE PENELITIAN**

Dalam bab ini, penulis membahas mengenai analisis dari sistem yang akan dibangun dan rancangan tampilan atau *interface* dari media pembelajaran yang akan dibangun.

**BAB IV HASIL DAN PEMBAHASAN**

Dalam bab ini, penulis memberikan penjelasan mengenai hasil dari media pembelajaran yang sudah selesai dibangun oleh penulis serta penjelasan terhadap hasil dari tampilan media pembelajaran yang sudah selesai dibangun.

**BAB V KESIMPULAN DAN SARAN**

Dalam bab ini, penulis memberikan beberapa kesimpulan serta saran yang berhubungan dengan pembahasan dan perancangan yang telah dilakukan dengan tujuan untuk dapat dijadikan sebagai panduan pada pengembangan berikutnya.

**BAB II**

**TINJAUAN PUSTAKA**

### Internet

Menurut Kamus Umum Bahasa Indonesia (KBBI), Internet merupakan jaringan komunikasi elektronik yang menghubungkan jaringan komputer dan fasilitas komputer yang terorganisasi di seluruh dunia melalui telepon atau satelit[15]. internet dimulai pada era 1960-an saat proyek *Advanced Research Projects Agency Network* (ARPANET) diinisiasi oleh Departemen Pertahanan Amerika Serikat. ARPANET adalah jaringan pertama yang menggunakan konsep paket data untuk menghubungkan beberapa komputer. Pada tahun 1969, ARPANET berhasil menghubungkan empat universitas di Amerika Serikat, membuka jalan bagi pertukaran informasi elektronik. Kemudian, pada tahun 1970-an, protokol TCP/IP (*Transmission Control Protocol/Internet Protocol*) diciptakan, menjadi dasar dari sistem komunikasi yang memungkinkan berbagai jaringan komputer untuk saling terhubung. Pada tahun 1980-an, jaringan ARPANET berkembang pesat, dan produsen komputer mulai memproduksi perangkat keras dan perangkat lunak yang mendukung penggunaan internet secara lebih luas.

Era 1990-an ditandai dengan perkembangan yang luar biasa dalam sejarah internet. *World Wide Web* (WWW) diciptakan oleh Tim Berners-Lee pada tahun 1991, memperkenalkan konsep *hyperlink* dan membuat internet lebih *user-friendly*. Ini menjadi landasan bagi pertumbuhan pesat penggunaan internet di seluruh dunia. Selanjutnya, layanan email, browsing, dan transfer file menjadi lebih umum di kalangan publik. Perusahaan-perusahaan teknologi seperti Google, Amazon, dan eBay mulai berkembang di tahun-tahun ini, mendorong ekspansi internet sebagai pusat perdagangan dan sumber informasi. Saat ini, internet telah menjadi bagian integral dari kehidupan manusia, mengubah cara komunikasi, perdagangan, dan akses informasi di seluruh dunia.

Saat ini Internet tidak hanya dapat digunakan untuk pertukaran informasi , bahkan internet juga dimanfaatkan untuk mengendalikan kendali pada benda-benda secara jarak jauh, dengan hanya tetap terkoneksi oleh internet yang biasa dikenal dengan *Internet of Things* (Iot), hal ini yang menjadi awal mula adanya revolusi industri 4.0.

### Internet of Things

*Internet of Things* adalah suatu penemuan yang dikembangkan karena memiliki kelebihan dari segi fungsionalitas dan mendukung untuk melakukan pekerjaan tanpa bantuan kabel atau hanya dengan melalui jaringan[8]. IoT merupakan jaringan yang menghubungkan perangkat non-komputer seperti perangkat rumah, kendaraan yang memiliki sensor, *software*, dan konektivitas sehingga terhubung ke internet, dan saling bertukar data[9]. *Internet of Things* (IoT) adalah sebuah konsep atau program yang mana objeknya memiliki kemampuan untuk mengirimkan atau mentransmisikan data melalui jaringan tanpa perlu bantuan perangkat komputer maupun bantuan dari manusia

*Internet of Things* (IoT) memiliki sejarah yang berakar pada konsep konektivitas antara perangkat elektronik. Meskipun istilah "*Internet of Things*" pertama kali muncul pada tahun 1982, pengembangan konsep ini berlangsung lebih intensif pada awal 2000-an ketika RFID (*Radio-Frequency Identification*) dan sensor mulai diimplementasikan untuk memungkinkan identifikasi dan pertukaran data antara perangkat tanpa perlu intervensi manusia. Sejak itu, terjadilah lonjakan penggunaan teknologi sensor, sistem monitoring jarak jauh, dan konektivitas internet yang memungkinkan perangkat-perangkat berkomunikasi dalam suatu jaringan. Perkembangan IoT semakin berkembang pesat dengan standar protokol komunikasi seperti

1. MQTT (*Message Queuing Telemetry Transport*): MQTT adalah protokol komunikasi ringan yang dirancang untuk kondisi jaringan yang tidak stabil. Protokol ini sangat efisien dan banyak digunakan dalam proyek IoT untuk mentransmisikan data sensor ke server atau antar perangkat[11].
2. CoAP (*Constrained Application Protocol*): CoAP adalah protokol desain khusus untuk perangkat terbatas seperti sensor IoT yang memiliki sumber daya terbatas, seperti daya dan *bandwidth*. CoAP biasanya digunakan untuk komunikasi antara perangkat IoT dan server[13].

yang memungkinkan perangkat untuk berbagi data secara efisien dalam ekosistem yang semakin terhubung.

### Sistem Keamanan

Pengertian Keamanan menurut KBBI adalah keadaan bebas dari bahaya. Istilah ini bisa digunakan dengan hubungan kepada kejahatan, segala bentuk kecelakaan, dan lain-lain[15]. Keamanan merupakan topik yang luas termasuk keamanan nasional terhadap serangan teroris, keamanan komputer terhadap *hacker* atau *cracker*, keamanan rumah terhadap maling dan penyelusup lainnya, keamanan finansial terhadap kehancuran ekonomi dan banyak situasi berhubungan lainnya

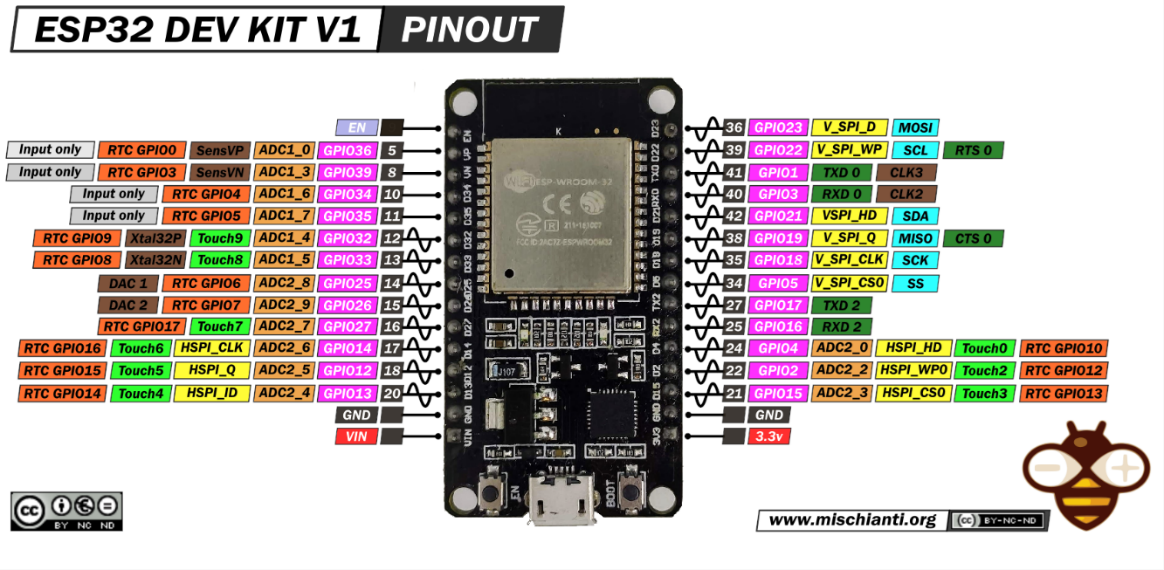
Sistem keamanan adalah serangkaian prosedur, teknologi, dan praktik yang dirancang untuk melindungi suatu entitas, seperti jaringan komputer, perangkat lunak, data, informasi rahasia, atau fisik dari ancaman yang mungkin merusak, merusak, atau mengganggu keamanan. Tujuan dari sistem keamanan adalah untuk mencegah akses yang tidak sah, melindungi data sensitif, mencegah serangan, dan mempertahankan integritas serta kerahasiaan sistem atau informasi[10].

Sistem keamanan dapat meliputi berbagai elemen, termasuk enkripsi data, *firewall*, perangkat lunak anti-virus, akses terenkripsi, pengamanan fisik seperti kunci, kartu akses, serta prosedur keamanan yang terkait dengan manajemen akses, pengawasan, dan pemantauan secara terus-menerus. Selain itu, pendidikan dan kesadaran akan keamanan juga penting, karena banyak ancaman keamanan terjadi karena kesalahan manusia.

### Mikrokontroler ESP32-CAM

MikroKontroler ESP32-CAM adalah sebuah sensor inovatif yang menggabungkan mikrokontroler kuat ESP32 dan modul kamera yang memungkinkan penggunaan teknologi WiFi serta kemampuan fotografi dalam sebuah perangkat yang kompak. Dikembangkan oleh Espressif Systems, sensor ini memberikan solusi yang efektif dalam proyek-proyek *Internet of Things* (IoT) dan keamanan rumah. Dengan modul kamera OV2640 berkekuatan 2MP, sensor ini mampu untuk mengambil gambar dan video dalam berbagai skenario aplikasi. Kelebihan utamanya terletak pada koneksi nirkabel WiFi 802.11 b/g/n dan Bluetooth v4.2, yang memungkinkan pengiriman data gambar dan video secara langsung atau remote. Ini menjadikannya pilihan populer bagi pengembang, hobiis, dan pemula yang ingin menyelidiki proyek-proyek yang memerlukan fungsi kamera dan konektivitas.

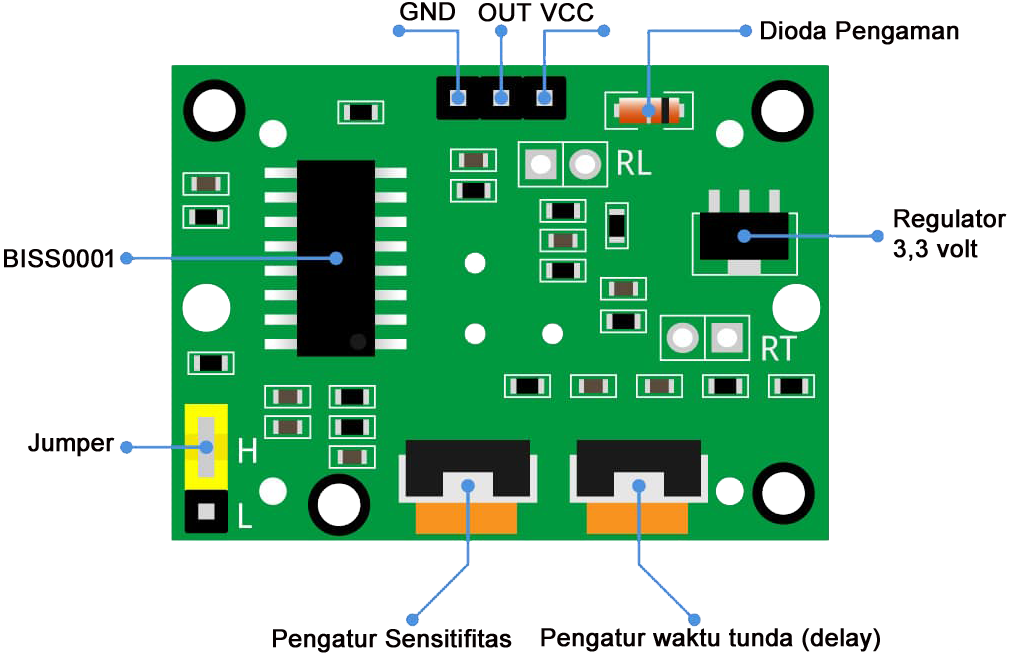
MikroKontroler ESP32-CAM merupakan solusi komprehensif bagi proyek-proyek yang membutuhkan integrasi kamera ke dalam platform IoT. Sensor ini tidak hanya memberikan akses ke teknologi WiFi yang canggih, tetapi juga memungkinkan akses ke antarmuka mikrokontroler yang luas seperti GPIO, UART, SPI, dan I2C, seperti yang terlihat pada Gambar 2.1 memfasilitasi kontrol yang lebih baik dalam aplikasi yang melibatkan sensor. Meskipun sensor ini membuka berbagai peluang kreatif dalam pembuatan sistem pengawasan, pemantauan, dan proyek-proyek visual lainnya, tetapi kualitas gambar atau video mungkin terbatas oleh sensor kamera terintegrasi. Namun, dengan dokumentasi yang luas dari *Espressif Systems* dan dukungan komunitas yang kuat, ESP32-CAM menawarkan solusi yang menarik bagi para pengembang yang ingin mengeksplorasi kemungkinan aplikasi berbasis kamera dalam proyek-proyek.

**Gambar 2.1 Skematik Mikrokontroller ESP32**

### Sensor PIR

Sensor PIR (*Passive Infrared*) merupakan komponen kritis dalam banyak aplikasi keamanan, otomatisasi, dan sensori. Sensor ini mendeteksi perubahan radiasi inframerah yang dihasilkan oleh pergerakan objek, seperti manusia atau hewan, dalam area deteksinya. Penggunaan sensor PIR meluas dalam sistem keamanan rumah, lampu otomatis, perangkat pintar, dan aplikasi IoT. Sebagian besar sensor PIR menggunakan prinsip kerja terhadap perubahan radiasi inframerah yang dipancarkan oleh objek yang memancarkan panas, menghasilkan sinyal yang memicu respon dalam sistem terkait. Sensor ini memainkan peran penting dalam menciptakan lingkungan yang lebih cerdas dan aman dengan mengenali pergerakan, yang kemudian digunakan untuk mengaktifkan atau menonaktifkan perangkat, atau untuk memberikan peringatan terhadap kejadian yang terdeteksi lihat seperti yang terlihat pada Gambar 2.2 .

Sensor PIR telah menjadi bagian integral dalam industri keamanan dan aplikasi otomatisasi. Dengan responsivitasnya terhadap pergerakan dan kemampuannya untuk mendeteksi perubahan suhu, sensor ini menghadirkan keuntungan besar dalam hal efisiensi energi dan keamanan, serta kenyamanan dalam lingkungan rumah atau perkantoran. Meskipun sensitivitasnya terhadap pergerakan objek dan lingkungan sekitarnya sangat diandalkan, ada juga beberapa kendala seperti ketidakmampuannya dalam membedakan antara sumber pergerakan, sehingga sensor ini rentan terhadap pemicu palsu. Namun, inovasi dan pengembangan dalam teknologi sensor PIR terus berlanjut, menjadikannya salah satu pilihan yang populer dalam aplikasi-aplikasi sensori yang mengandalkan deteksi gerak untuk berbagai tujuan, baik keamanan maupun otomatisasi.

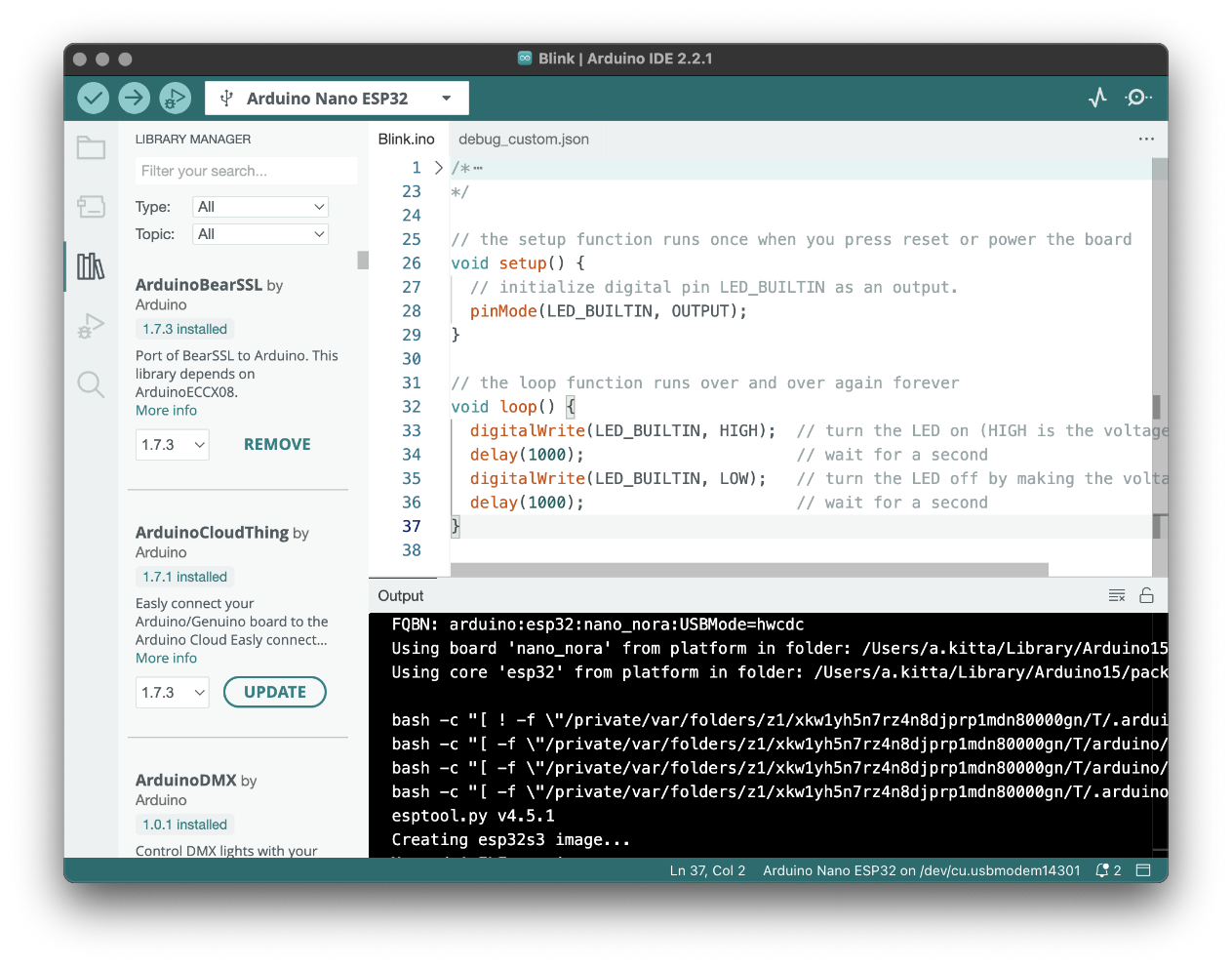


**Gambar 2.2 Skematik Sensor PIR HC-SR501**

### Arduino IDE

Pengertian Arduino IDE mencakup peran krusial dalam pengembangan sistem keamanan berbasis IoT dengan mikrokontroler ESP32. Arduino IDE, atau *Integrated Development Environment*, dapat didefinisikan sebagai lingkungan pengembangan perangkat lunak yang khusus dirancang untuk memprogram dan mengelola mikrokontroler, termasuk yang digunakan dalam proyek IoT keamanan. Kelebihan utama Arduino IDE terletak pada antarmuka yang mudah digunakan, menjadikannya pilihan ideal, terutama bagi pemula dalam pemrograman mikrokontroller

Instalasi dan konfigurasi Arduino IDE untuk ESP32 merupakan langkah awal yang krusial dalam pengembangan sistem keamanan. Panduan ini memastikan bahwa lingkungan pengembangan siap digunakan untuk proyek keamanan kantor yang melibatkan kontrol akses dan pemantauan berbasis IoT. Pemrograman mikrokontroler dengan Arduino IDE melibatkan penulisan kode yang terstruktur, memungkinkan integrasi sensor keamanan dengan mudah, serta kontrol akses yang handal seperti yang terlihat pada Gambar 2.3. Manajemen perangkat lunak dan *debugging* dalam Arduino IDE memberikan alat bantu efektif bagi pengembang. Melalui fitur ini, pengguna dapat mengelola perpustakaan fungsionalitas untuk mendukung implementasi sensor keamanan, sementara alat debugging memudahkan pengidentifikasian dan perbaikan kesalahan dalam kode program.

Dengan demikian, dalam pengembangan sistem keamanan berbasis IoT, pemahaman dan penerapan Arduino IDE menjadi langkah kunci untuk merancang solusi keamanan yang efektif, dapat diandalkan, dan sesuai dengan kebutuhan proyek spesifik.

**Gambar 2.3 Interface Arduino IDE 2.2.1**

### Penelitian Terdahulu

Hal yang sangat penting dalam penelitian saat ini adalah dasar atau landasan teori-teori dari hasil penelitian sebelumnya atau penelitian terdahulu dan hasil dari penelitian sebelumnya tersebut dapat dijadikan sebagai data pendukung. Data yang digunakan oleh peneliti sebagai data pendukung harus memiliki kaitan dengan permasalahan yang dibahas oleh peneliti saat ini. Penelitian ini didasarkan pada penelitian terdahulu dengan topik yang memiliki kesamaan dengan penelitian peneliti saat ini Adapun penelitian terdahulu untuk menjadi rujukan metodologi penelitian ini sebagai berikut

Perancangan Robot Monitoring Keamanan Kantor Menggunakan MikroKontroler ESP32CAM Berbasis *Internet Of Things* bertujuan untuk membuat robot *monitoring* dan *controlling* yang berfungsi mengawasi suatu ruangan kantor dengan melengkapi modul kamera yang terdapat pada Mikrokontroler ESP32cam dan robot ini mengirimkan gambar secara realtime ke web server robot juga dapat dikendalikan melalui dua perangkat yaitu *laptop* atau *smartphone* dan melalui *joystick*. Dari penelitian pengembangan robot keamanan kantor berbasis IoT dengan ESP32CAM, dapat disimpulkan bahwa board ESP32CAM berhasil mengirimkan data foto ke web server melalui protokol HTTP menggunakan jaringan nirkabel. Robot memiliki tiga mode gerak, termasuk otomatis, *joystick*, dan WiFi, sementara sensor ultrasonik dan PIR berfungsi untuk mengukur jarak dan mendeteksi keberadaan manusia[5]

Sistem Keamanan Pada Ruangan Pribadi Menggunakan Mikrokontroler Arduino Dan Sms *Gateway* bertujuan merancang suatu sistem keamanan rumah yang terdiri dari sensor PIR sebagai sensor untuk mendeteksi pencuri dan sensor Magnet sebagai indikator pintu atau jendela apakah pintu atau jendela sedang ditutup atau terbuka, mikrokontroler arduino UNO sebagai proses atau otak dari alat dan SIM 800L sebagai transmiter pesan notifikasi kepada pemilik rumah yang berupa sms berbasis mikrokontroler, sehingga dapat benar-benar membantu seseorang baik mengenai efisiensi biaya dan waktu dalam memonitoring keadaan rumah[10].

Dari hasil pengujian yang telah dilakukan maka dapat ditarik kesimpulan bahwa:

1. Komponen yang digunakan dalam pembuatan alat berupa Arduino UNO, sensor magnet, sensor PIR, SIM 800L, *buzzer*, dan akrilik yang digunakan sebagai miniatur kerangka dari ruangan dan dapat berfungsi dengan baik
2. Pembuatan alat keamanan ruangan pribadi menggunakan sensor magnet yang diletakan di bagian pintu ruangan yang dimana sensor magnet terhubung ke mikrokontroler serta dapat menginformasikan melalui SIM 800L apabila sensor magnet mendapatkan data bahwa pintu terbuka
3. Sensor PIR dapat berfungsi sebagai pendeteksi pergerakan pada seseorang yang ada didalam ruangan pribadi dengan jarak akurat sampai dengan 5 meter
4. SIM 800L berfungsi sebagai pengirim dari keadaan yang ada diruangan pribadi apakah ada sebuah pergerakan atau pintu dalam kondisi terbuka
5. Ketika sensor PIR dan sensor magnet mendapat kan kondisi maka SIM 800L akan aktif dan mengirimkan informasi ke pemilik ruangan pribadi tersebut.

Pengembangan Smart Home Dengan Mikrokontroler ESP32 Dan MC-38 *Door Magnetic Switch* Sensor Berbasis *Internet of Things* (IoT) Untuk Meningkatkan Deteksi Dini Keamanan Perumahan bertujuan untuk memanfaatkan Smart Home dengan microcontroller ESP32 terintegrasi dengan kamera dan MC-38 *door magnetic* switch sensor berbasis *Internet of Things* (IoT) sebagai dasar penelitian untuk mendeteksi keamanan perumahan bumi Arumsari Sumber Kabupaten Cirebon pada saat ditinggal oleh penghuninya[11].Berdasarkan hasil dari pengujian dan pembahasan diperoleh data dan fakta, untuk sudut 0 derajat sampai sudut 5 derajat, ESP32-CAM dapat merespon gambar dan memberikan gambar melalui arduino yang kemudian ditampilkan melalui jendela browser, dengan status MC-38 *door magnetic* switch sensor normally close dan normally open. Sedangkan untuk sudut diatas 5 derajat sampai dengan 180, ESP32-CAM merespon gambar dan memberikan gambar melalui arduino, kemudian ditampilkan melalui jendela browser, dengan status MC-*38 door magnetic switch* sensor *normally open*. Hal ini kemungkinan disebabkan adanya gaya lorentz yang ditimbulkan adanya medan listrik, dan bergerak melalui suatu medan magnet

Pemanfaatan ESP32 Pada Sistem Keamanan Rumah Tinggal Berbasis IoT bertujuan untuk membuat sistem keamanan yang dapat mendokumentasikan aktifitas yang terjadi dalam ruangan Rumah tinggal dengan memanfaatkan kontroler Esp32 berbasis IoT[13]. Sistem yang dibuat dilengkapi dengan sensor PIR yang difungsikan untuk mendeteksi orang yang berada dalam ruangan dan secara otomatis mengaktifkan alaram dan kamera untuk melakukan proses pengambilan gambar . Dari hasil penelitian yang dilakukan maka dapat disimpulkan bahwa Sistem keamanan rumah tinggal dapat melakukan proses pengamanan, melalui pendeteksian objek manusia dalam ruangan melalui sensor PIR, dengan sensitifitas sensor berkisar antara 1 sampai dengan 4 meter. Saat terdeteksi ada objek manusia dalam ruangan, maka sistem akan mengaktifkan web cam guna pengambilan gambar dan mengirimkan data gambar tersebut ke web servermelalui komunikasi wifi.

Penerapan Sistem Keamanan Gerbang Rumah Berbasis Telegram Menggunakan Esp8266 Yudha bertujuan untuk membantu mengontrol pintu gerbang rumah masyarakat dari tindakan criminal dan mampu memberikan notifikasi pada telegram pada saat gerbang tidak tertutup. Setelah dilakukan berbagai pengujian dan analisa pada alat atau sistem gerbang otomatis baik itu perangkat keras maupun perangkat lunak[14]. Maka dapat diambil kesimpulan sebagai berikut:

1. Perancangan sistem keamanan gerbang dilakukan dengan beberapa tahapan mulai dari pembuatan blok diagram,perancangan rangkaian ,*flowchart*, membuat rangkaian skematik alat dan mengimplementasikanya ke perangkat keras dan perangkat lunak untuk diambil data lalu dianalisis untuk mendapatkan kesimpulan.
2. Implementasi alat ini dilakukan dengan cara menguji coba alat terlebih dahulu selama 10 hari untuk mengetahui kelayakan pada alat setelah itu alat di implementasikan pada gerbang kosan yang menjadi tempat penelitian. Pada aplikasi telegram ditanamkan bot yang dibutuh kan dan menggunakan wifi yang terhubung.

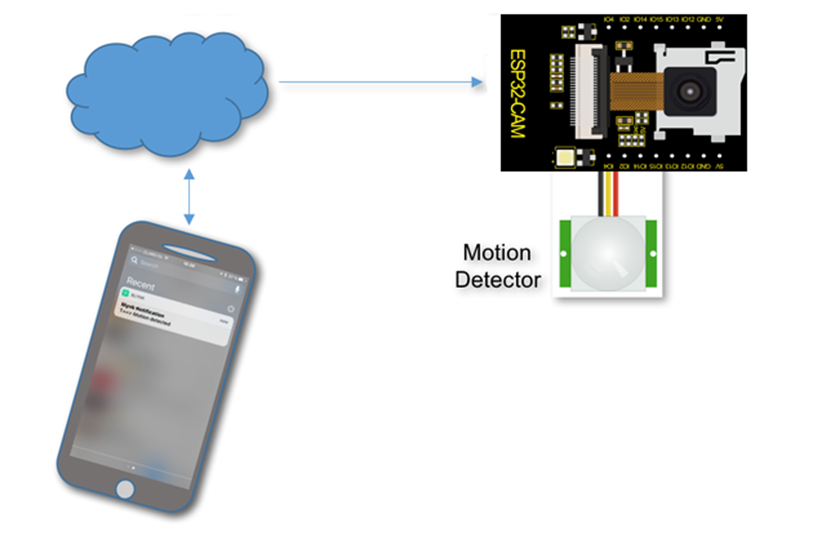
**BAB III**

**METODE PENELITIAN**

#### Analisis

Pada analisis sistem, penulis melakukan analisis Masalah yang akan dibahas penelitian ini adalah kejahatan pada lingkungan perkantoran yang tiap tahun meningkat walaupun dalam kantor ada CCTV tetapi pada waktu malam pengawas CCTV ataupun pihak keamanan mungkin tidak terlalu memperhatikan ataupun focus karena adanya faktor seperti tingkat kewaspadaan rendah. Hal ini terjadi dikarenakan kebanyakan CCTV belum memiliki kemampuan untuk melaporkan adanya pergerakan pada waktu malam sehingga pengawas CCTV tidak fokus dalam mengawasi kantor.

Sistem keamanan ini dirancang dengan menggunakan mikrokontroler ESP32 sebagai otak utama yang mengkoordinasikan seluruh operasi, di mana ESP32 ini terhubung langsung ke Sensor PIR (*Passive Infrared Sensor*) yang berfungsi untuk mendeteksi gerakan dalam area yang diawasi. Ketika Sensor PIR mendeteksi adanya gerakan, sinyal akan dikirim ke ESP32, yang kemudian mengaktifkan protokol komunikasi melalui Wi-Fi untuk terhubung ke *platform* *Blynk*. Dalam platform *Blynk*, sebuah dashboard khusus telah dikonfigurasikan untuk menerima dan menampilkan notifikasi serta data terkait deteksi gerakan, yang memungkinkan pengguna untuk memantau keamanan kantor secara real-time dari jarak jauh melalui aplikasi *Blynk* di smartphone mereka. Interaksi ini tidak hanya sebatas penerimaan data, tetapi juga pengguna dapat mengirim perintah kembali ke ESP32 melalui *Blynk* untuk mengendalikan sistem, seperti mengaktifkan atau menonaktifkan sensor atau perangkat lain yang terhubung, sehingga menciptakan sistem keamanan yang interaktif dan responsif terhadap kebutuhan keamanan kantor terlihat pada Gambar 3.1 dapat dilihat dari diagram alur tentang cara dasar sistem ini bekerja :



3

2

1

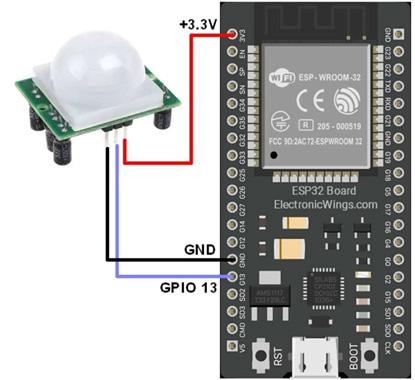
**Gambar 3.1 Diagram Arsitektur Sistem**

Keterangan

1. Koneksi Sensor PIR ke ESP32:

Pada Gambar 3.1, dapat terlihat representasi skematik dari infrastruktur dasar sistem yang menggambarkan hubungan antara WiFi dan aplikasi. Gambar ini secara visual menguraikan bagaimana data dan perintah bergerak dari pengguna melalui jaringan WiFi untuk diakses dan dikelola oleh aplikasi. Hal ini menetapkan landasan bagi interaksi antarmuka pengguna dengan sistem dan merupakan langkah awal dalam proses komunikasi antara pengguna dan hardware melalui aplikasi.

Melangkah lebih jauh dari dasar komunikasi tersebut, Gambar 3.2 akan menggali lebih dalam ke dalam mekanisme operasional hardware itu sendiri. Ini akan menyoroti bagaimana aplikasi, setelah menerima dan memproses informasi dari pengguna melalui WiFi, mengkomunikasikan instruksi tersebut ke hardware. Dengan demikian, memungkinkan untuk memahami secara lebih komprehensif bagaimana perangkat keras merespons dan menjalankan tugas-tugas yang diminta berdasarkan input yang diterima dari aplikasi



a

b

c

**Gambar 3.2 Skematik Sensor PIR dan Mikrokontroler ESP32**

Keterangan

1. Pin GND dari sensor dihubungkan ke pin GND pada ESP32 untuk menyelesaikan sirkuit.
2. Pin data dari sensor PIR dihubungkan ke salah satu pin GPIO pada ESP32 yang dikonfigurasi sebagai output digital.
3. Pin Vcc dari sensor PIR dihubungkan ke pin 3V3 pada ESP32 untuk menyediakan daya.
4. Koneksi ESP32 ke Internet:

ESP32 harus dikonfigurasi untuk terhubung ke jaringan Wi-Fi melalui kode program. Ini memungkinkan ESP32 mengakses internet dan berkomunikasi dengan *platform* *Blynk*. Dalam konfigurasi perangkat lunak: Menggunakan Arduino IDE atau *platform* IO, instalasi pustaka *Blynk* dan pustaka untuk ESP32 perlu dilakukan. Kode harus meliputi:

1. Konfigurasi Wi-Fi: Inisialisasi koneksi Wifi menggunakan SSID dan password jaringan.
2. Inisialisasi *Blynk*: Menggunakan *Blynk* Auth Token yang didapatkan dari aplikasi *Blynk*.
3. Setup sensor PIR: Konfigurasi pin GPIO yang terhubung ke sensor PIR sebagai input.
4. Loop utama: Membaca input dari sensor PIR dan menggunakan *Blynk*.*notify*() untuk mengirim notifikasi saat deteksi gerakan.
5. Koneksi Internet ke *Blynk*

Setelah ESP32 berhasil terhubung ke Wi-Fi dan terintegrasi dengan kode program yang telah disiapkan, proses komunikasi antara ESP32 dan server *Blynk* melalui internet dapat dimulai. Proses ini melibatkan beberapa langkah penting yang memungkinkan data dan perintah untuk berpindah dari perangkat fisik Anda ke aplikasi *Blynk* di smartphone.

1. Pengiriman Data ke Server *Blynk*
2. Autentikasi: Pertama-tama, ESP32 melakukan autentikasi dengan server *Blynk* menggunakan *Blynk Auth Token* yang telah diinisialisasi di dalam kode program. Token ini unik untuk setiap proyek *Blynk* dan berfungsi sebagai kunci akses yang memungkinkan komunikasi yang aman antara hardware dan server *Blynk*.
3. Koneksi: Setelah autentikasi berhasil, ESP32 terhubung ke server *Blynk*. Proses ini menggunakan protokol internet standar (seperti HTTP/HTTPS atau protokol MQTT yang lebih ringan) untuk memungkinkan pertukaran data secara real-time.
4. Pengiriman Data: ESP32 kemudian secara berkala mengirimkan data dari sensor PIR ke server *Blynk*. Data ini bisa berupa status deteksi gerakan (misalnya, "gerakan terdeteksi" atau "tidak ada gerakan"). Proses ini terjadi melalui fungsi loop utama dalam kode program, di mana ESP32 terus memantau input dari sensor PIR.
5. Proses di Server *Blynk*
6. Penerimaan Data: Server *Blynk* menerima data dari ESP32 dan memprosesnya sesuai dengan konfigurasi proyek yang telah dibuat di aplikasi *Blynk*. Server ini bertindak sebagai perantara antara hardware dan aplikasi pengguna.
7. Logika Aplikasi: Berdasarkan data yang diterima, logika aplikasi pada server *Blynk* mengevaluasi apakah perlu mengirim notifikasi. Misalnya, jika data dari sensor PIR menunjukkan deteksi gerakan, server akan memicu proses pengiriman notifikasi.
8. Pengiriman Notifikasi ke Aplikasi *Blynk*
9. Trigger Notifikasi: Jika kondisi untuk pengiriman notifikasi terpenuhi (seperti deteksi gerakan oleh sensor PIR), server *Blynk* mengirim perintah notifikasi ke aplikasi *Blynk* yang terinstal di smartphone pengguna.
10. Penerimaan Notifikasi: Aplikasi *Blynk* pada smartphone pengguna menerima notifikasi dan menampilkannya. Pengguna dapat melihat notifikasi ini secara real-time, yang memberikan informasi langsung tentang aktivitas yang terdeteksi oleh sensor PIR.

Setup *Blynk* Dalam aplikasi *Blynk*:

1. Buat proyek baru dan pilih ESP32 sebagai perangkat.
2. Salin *Blynk Auth Token* yang diberikan ke dalam kode program ESP32.
3. Tambahkan widget notifikasi untuk menerima alert pada smartphone.

Integrasi dan Pengujian:

1. Setelah semuanya terhubung, uji sistem dengan menggerakkan tangan di depan Sensor PIR.
2. Periksa apakah notifikasi muncul pada smartphone melalui aplikasi *Blynk*.

Sistem yang diusulkan dalam penelitian ini adalah pengembangan aplikasi untuk memudahkan pihak keamanan dalam mengawasi ruangan kantor serta memberikan informasi *real-time* kepada mereka. Sistem ini dirancang untuk meningkatkan efisiensi dan responsivitas dalam situasi keamanan melalui implementasi teknologi *Internet of Things* (IoT) menggunakan mikrokontroler ESP32. Dengan integrasi sensor PIR (*Passive Infrared Sensor*) untuk deteksi gerakan pada malam hari, sistem ini akan mengaktifkan mikrokontroler ESP32 untuk mengambil gambar ketika aktivitas terdeteksi pada waktu malam. Gambar tersebut kemudian dikirim ke pengawas CCTV sambil menyimpan data terkait ke dalam basis data untuk keperluan audit dan analisis keamanan lebih lanjut. Melalui pendekatan ini, aplikasi bertujuan untuk memberikan solusi keamanan yang proaktif dengan memanfaatkan kecanggihan teknologi IoT.

Sistem harus mampu menjamin keamanan pada ruangan kantor yang terpasang alat. Selain itu, sistem perlu dirancang untuk dapat bekerja secara kontinu dan stabil dalam jangka waktu yang lama, dengan meminimalisir waktu henti akibat perawatan atau gangguan. Dalam proses pemodelan sistem, akan ditentukan arsitektur dan desain operasional sistem secara rinci. Pemodelan ini akan menggunakan *UML* (*Unified Modeling Language*) untuk merepresentasikan struktur dan perilaku sistem, termasuk skenario interaksi antara sistem dengan lingkungan sekitarnya dan antar komponen sistem. Pemodelan ini menggunakan *UML* *Flowchart* metode adalah gambaran visual yang digunakan untuk mengilustrasikan langkah-langkah berurutan dalam sebuah metode atau proses. *Flowchart* ini menunjukkan jalur logis dari langkah-langkah yang harus dijalankan untuk mencapai hasil yang ditampilkan pada Gambar 3.3

Mulai

Kirim data ke pihak keamanan

Sensor PIR mendeteksi adanya gerakan

Notifikasi pihak keamanan

Mengaktifkan mikrokontroler ESP32

selesai

Mikrokontroler ESP32 mengambil gambar

Simpan data gambar

**Gambar 3.3 *Flowchart* Metode**

#### Perancangan

Adapun perancangan sistem dapat dilihat antara lain:

Perancangan *Interface* utama adalah tampilan yang pertama kali muncul ketika aplikasi dijalankan, pada interface utama terdapat beberapa tampilan seperti terlihat pada Gambar 3.3 dibawah ini

*1.On/Off*

2.. *Log Alert*

**Gambar 3.3 Rancangan *Interface* Utama**

Keterangan

1. Merupakan *Switch Button* berfungsi untuk mengaktifkan/menonaktifkan perangkat
2. Merupakan *List* berfungsi untuk menampilkan tampilan daftar catatan pemberitahuan yang telah dikirimkan

Dalam rangka merancang sistem deteksi gerakan yang efisien dan responsif, di bawah ini disajikan *pseudocode* yang menggambarkan algoritma utama sistem. *Pseudocode* ini dimaksudkan untuk memberikan gambaran kasar tentang proses kerja sistem, mulai dari inisialisasi perangkat keras yang meliputi konfigurasi kamera dan konektivitas WiFi, hingga pengolahan input dari sensor PIR dan pemicu pengambilan foto. Pseudo code ini bertujuan untuk memudahkan pemahaman logika operasional sistem sebelum implementasi kode sebenarnya dilakukan dalam bahasa pemrograman yang dipilih.Dengan menggunakan pendekatan ini, kita dapat merinci langkah-langkah yang diperlukan untuk mencapai fungsi utama sistem, yang mencakup:

1. Konfigurasi Awal: Menyiapkan perangkat keras dan variabel yang diperlukan untuk operasi sistem.
2. Koneksi WiFi dan *Blynk*: Menghubungkan sistem ke internet dan platform *Blynk* untuk memungkinkan komunikasi jarak jauh.
3. Deteksi Gerakan: Menanggapi input dari sensor PIR untuk mendeteksi keberadaan gerakan.
4. Pengambilan dan Pengiriman Foto: Mengambil foto melalui kamera yang terintegrasi dan mengirimkan data tersebut ke pengguna melalui *Blynk*.

Pemahaman mendalam terhadap algoritma ini penting sebelum melanjutkan ke tahap pengembangan perangkat lunak, untuk memastikan bahwa semua fungsi sistem dapat diimplementasikan secara efektif dan efisien. Berikut adalah *pseudocode* yang merangkum proses tersebut:

Langkah-langkah dalam konfigurasi :

1. Inisialisasi Konfigurasi Kamera dan Koneksi WiFi
2. Mulai dengan konfigurasi serial.
3. Atur pin *LED* sebagai output.
4. Konfigurasikan kamera:
5. Tentukan *channel, timer*, pin, format piksel, ukuran *frame*, kualitas JPG, dan jumlah *buffer* berdasarkan keberadaan PSRAM.
6. Inisialisasi kamera dan periksa kesalahan.
7. Jika sensor kamera adalah OV 3660, atur flip vertikal, kecerahan, dan saturasi.
8. Turunkan ukuran frame untuk meningkatkan frame rate awal.
9. Sambungkan ke WiFi menggunakan SSID dan *password* yang diberikan.
10. Tunggu sampai koneksi berhasil.
11. Mulai server kamera.
12. Tampilkan alamat IP kamera yang siap digunakan.
13. Mulai koneksi *Blynk* dengan auth token dan detail WiFi.
14. Loop Utama
15. Jalankan *Blynk* secara berulang.
16. Jika sensor PIR mendeteksi gerakan (*LOW*):
17. Cetak pesan notifikasi ke serial.
18. Kirim notifikasi "*Intruder Detected*..." menggunakan *Blynk*.
19. Cetak "*Capture Photo*" ke serial dan ambil foto.
20. Tunda selama 3 detik.
21. Jika tombol foto ditekan (*HIGH*):
22. Cetak "*Capture Photo*" ke serial dan ambil foto.
23. Fungsi Ambil Foto
24. Aktifkan *LED*, tunggu 200ms.
25. Buat nomor acak.
26. Cetak *URL* untuk mengakses foto yang diambil ke serial.
27. Kirim *URL* ke *Blynk* untuk ditampilkan.
28. Matikan *LED* dan tunggu 1 detik.

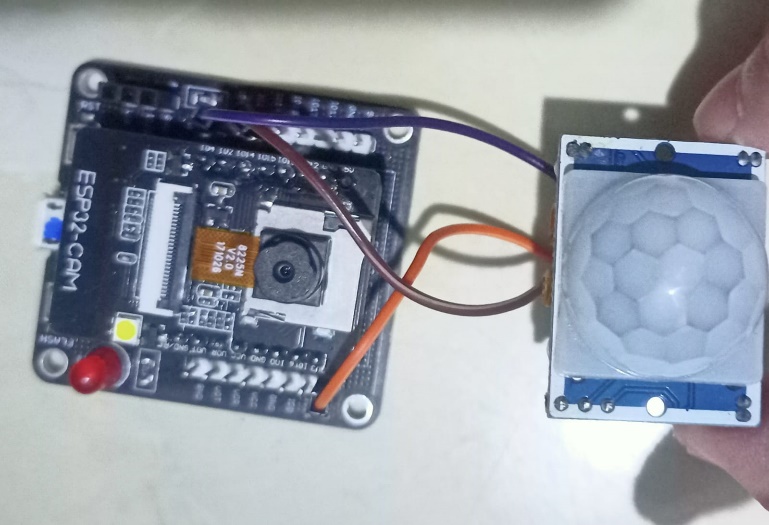
Melalui *pseudocode* ini, akan mendapatkan kerangka kerja awal yang akan digunakan sebagai dasar dalam pengembangan kode program yang sesungguhnya. Ini memungkinkan untuk penyesuaian dan optimasi lebih lanjut sesuai dengan kebutuhan spesifik proyek, sekaligus memastikan bahwa aspek kritis dari desain sistem telah dipertimbangkan sejak awal.

**BAB IV**

**HASIL DAN PEMBAHASAN**

##### Hasil

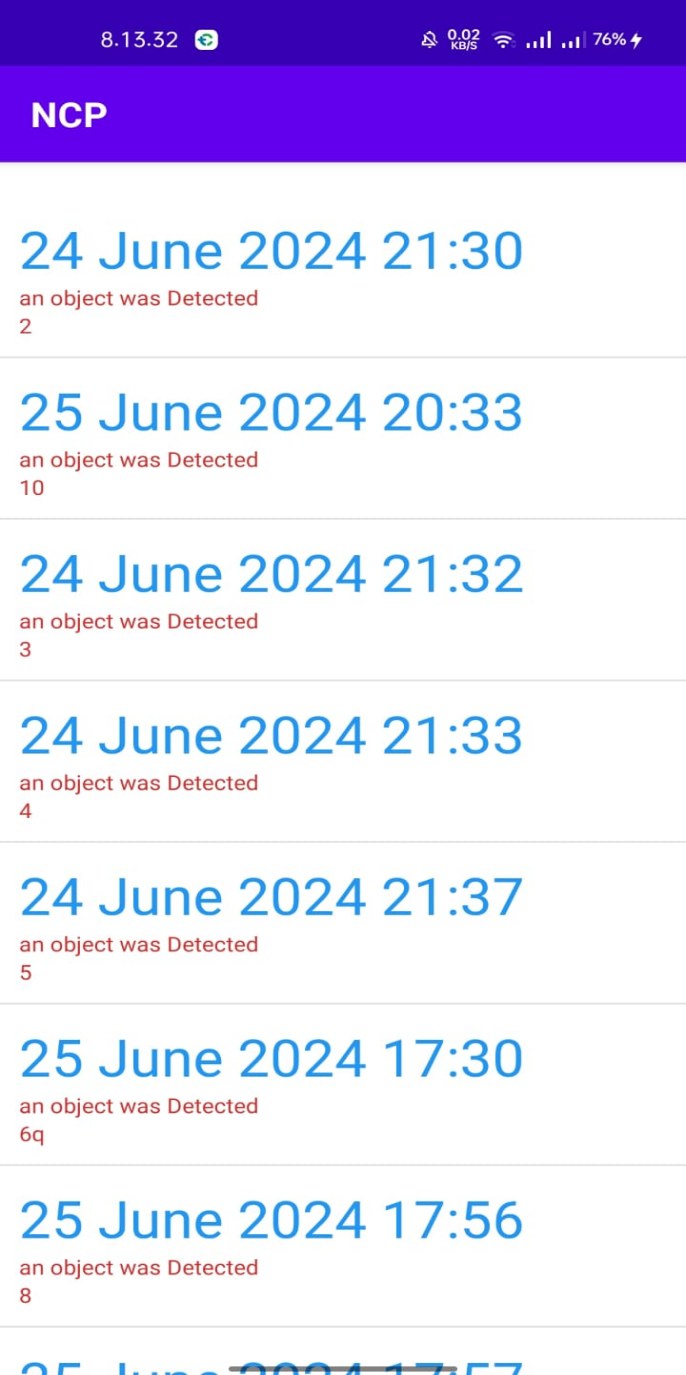
Pada tahap hasil sistem ini akan dibahas mengenari penerapan dan aplikasi dari hasil analisis dan perancangan sistem yang telah dipaparkan dalam bab sebelumnya , serta perangkat yang dibutuhkan untuk menjalankan aplikasi ini pada Gambar 4.1 terlihat detail penyambungan pin sensor pir ke ESP32



**Gambar 4.1 Rangkaian PIR-ESP32**

1. Halaman Utama

Halaman ini menunjukkan halaman welcome saat pemakai ataupun pengguna membuka aplikasi. Halaman utama ditunjukkan pada Gambar 4.2



##### 

**Gambar 4.2 Halaman Utama**

1. Halaman *Log* Gambar

Halaman ini menampilkan *log* gambar atau foto yang diambil oleh ESP32 CAM dan ditampilkan melalui telegram dapat dilihat dari Gambar 4.3



**Gambar 4.3 Halaman Log Gambar**

1. Pengujian sensor PIR

Pengujian sensor PIR bertujuan untuk mendapatkan data sehubungan dengan kerja sensor dalam mendeteksi pergerakan berdasarkan suhu tubuh manusia. Adapun data hasil pengujian sensor PIR seperti yang diperlihatkan pada Tabel 4.1

**Tabel 4.1 Pengujian Sensor PIR**

|  |  |  |
| --- | --- | --- |
| Objek | Jarak(Meter) | Respon |
| Manusia | 1 | Aktif |
| 2 | Aktif |
| 3 | Aktif |
| 4 | Aktif |
| 5 | Tidak Aktif |
| 6 | Tidak Aktif |
| 7 | Tidak Aktif |
| 8 | Tidak Aktif |

Dari data yang ditunjukan pada Tabel 4.1, terlihat bahwa sensitifitas sensor PIR tergantung pada jarak objek dan sensor , dimana sensor mendeteksi objek yang bergerak pada kisaran 1 sampai dengan 4 meter. Sedangkan pada jarak diatas 4 meter, sensor PIR tidak dapat mendeteksi objek. Untuk mendapatkan hasil yang maksimal pada proses keamanan kantor, maka penempatan sensor PIR. untuk mendeteksi pergerakan haruslah dipasang pada tempat-tempat yang dianggap berpotensi. Karena jarak jangkauan sensor PIR maksimal adalah 4 meter, maka jarak penempatan tiap-tiap sensor harus diatur agar dapat mendeteksi pergerakan dalam ruangan secara maksimal.

##### Pembahasan

Pada Program ESP32-CAM yang dikembangkan dengan menggunakan *Firebase Firestore* dan *Telegram Bot* dapat menunjukkan sebuah pendekatan yang sangat efisien dalam mengelola data serta berinteraksi dengan pengguna. Dengan memanfaatkan *Firebase Firestore* sebagai basis data utama, program ini mampu memanfaatkan keunggulan teknologi *database real-time* untuk memastikan respons yang cepat dan keakuratan data yang tersimpan. *Firebase Firestore* tidak hanya menyediakan kemampuan penyimpanan data yang skalabel, tetapi juga memungkinkan untuk sinkronisasi data secara otomatis antara perangkat ESP32-CAM dan platform cloud, sehingga informasi yang diperlukan pengguna dapat diakses dengan mudah dan dalam waktu nyata.

Penggunaan *Telegram Bot* sebagai antarmuka komunikasi utama merupakan pilihan yang strategis dalam program ini. *Telegram Bot* menawarkan platform yang stabil dan handal untuk pengiriman pesan dan *file*, seperti foto yang diambil oleh kamera ESP32-CAM. Dengan integrasi yang baik antara *Telegram Bot API* dan ESP32-CAM, pengguna dapat dengan mudah mengirim perintah melalui *Telegram*, menerima notifikasi, serta melihat hasil tanggapan dari perangkat secara langsung melalui aplikasi Telegram yang sudah umum digunakan. Hal ini memberikan pengalaman pengguna yang intuitif dan efisien, tanpa perlu instalasi aplikasi tambahan atau konfigurasi yang rumit.

Keputusan untuk mengganti *platform* seperti *Blynk* dengan pengembangan aplikasi Android memberikan fleksibilitas tambahan dalam desain antarmuka pengguna yang lebih sesuai dengan kebutuhan proyek. Meskipun aplikasi Android dapat memberikan kontrol yang lebih besar terhadap tampilan dan fungsionalitas, keputusan ini juga memerlukan investasi tambahan dalam waktu dan sumber daya untuk pengembangan serta pemeliharaan aplikasi. Dalam konteks ESP32-CAM yang mungkin memiliki keterbatasan dalam daya dan prosesor, *Telegram Bot* menawarkan solusi yang lebih ringkas dan efisien dalam hal kinerja serta integrasi.

Dari segi kinerja, program ini diatur untuk mengoptimalkan interaksi antara perangkat keras ESP32-CAM dengan pengguna melalui solusi yang efektif dan responsif dalam pengiriman data. Konfigurasi kamera dan sensor pada ESP32-CAM dioptimalkan untuk memastikan pengambilan gambar yang baik dengan kualitas yang memadai untuk keperluan pemantauan atau pengiriman informasi visual melalui *Telegram*. Proses pengiriman foto yang dilakukan dengan metode chunking memastikan bahwa pengiriman data berjalan lancar meskipun dalam kondisi koneksi yang terbatas.

Secara keseluruhan, integrasi antara ESP32-CAM, *Firebase Firestore*, dan *Telegram Bot* tidak hanya meningkatkan efisiensi dalam pengelolaan data dan interaksi pengguna, tetapi juga memberikan solusi yang tangguh dan dapat diandalkan dalam konteks aplikasi IoT. Penggunaan teknologi modern ini tidak hanya mempercepat proses komunikasi dan pengiriman data, tetapi juga memungkinkan adaptasi yang lebih baik terhadap kebutuhan dan tantangan teknis yang mungkin muncul dalam pengembangan aplikasi IoT pada masa depan. Dengan demikian, program ini tidak hanya menghadirkan inovasi dalam integrasi teknologi, tetapi juga mengoptimalkan pengalaman pengguna dalam penggunaan aplikasi IoT secara menyeluruh.

**4.2.1 Kelebihan Sistem Usulan:**

Adapun kelebihan sistem usulan yang dirancang oleh penulis antara lain sebagai berikut :

1. Integrasi dengan *Firebase Firestore*:

1. *Real-time Database*: *Firebase Firestore* menyediakan basis data yang real-time, memungkinkan sinkronisasi data secara otomatis antara perangkat ESP32-CAM dan *cloud Firebase*. Hal ini memastikan bahwa data yang disimpan dan diambil oleh perangkat selalu terkini dan akurat.
2. Skalabilitas: *Firebase Firestore* mudah diatur untuk menangani jumlah pengguna dan volume data yang besar tanpa memerlukan konfigurasi infrastruktur yang rumit. Ini membuat sistem dapat berkembang seiring dengan kebutuhan proyek tanpa mengalami kendala performa atau ketersediaan.
3. Keamanan: *Firebase* menyediakan kontrol akses yang kuat dan otentikasi pengguna terintegrasi, sehingga data sensitif yang disimpan di *cloud* tetap aman dari akses yang tidak sah atau perubahan yang tidak diinginkan.

2. Penggunaan *Telegram* *Bot*:

1. Interaksi Pengguna yang Mudah: Telegram Bot menawarkan antarmuka yang intuitif bagi pengguna untuk berinteraksi dengan perangkat ESP32-CAM. Pengguna dapat dengan mudah mengirim perintah, menerima notifikasi, dan melihat hasil dari perangkat melalui aplikasi *Telegram* yang sudah mereka kenal dan gunakan.
2. Ketersediaan Platform: *Telegram Bot* tersedia secara luas dan mendukung penggunaan di berbagai perangkat dan sistem operasi, termasuk desktop dan mobile. Ini memungkinkan fleksibilitas dalam akses dan penggunaan tanpa membatasi pengguna untuk menggunakan platform khusus.

3. Pergantian dari *Blynk*:

1. Biaya yang Efisien: Salah satu alasan utama pergantian dari *Blynk* adalah untuk menghindari biaya langganan ke fitur pro yang mungkin tidak terjangkau untuk penggunaan eksperimental atau pengujian seperti yang dilakukan pada proyek ini. *Firebase Firestore* dan *Telegram Bot* menyediakan fitur-fitur yang diperlukan tanpa memerlukan biaya berlangganan yang signifikan.
2. Fleksibilitas dalam Pengembangan: Menggunakan *Firebase Firestore* dan *Telegram Bot* juga memberikan fleksibilitas lebih besar dalam desain dan pengembangan antarmuka pengguna. Dibandingkan dengan *Blynk* yang mungkin membatasi dalam hal tata letak dan desain antarmuka, pengembang memiliki kontrol yang lebih besar terhadap bagaimana pengguna berinteraksi dengan perangkat

.

**4.2.2 Kelemahan Sistem Usulan:**:

Adapun kelemahan sistem usulan yang dirancang oleh penulis antara lain sebagai berikut:

1. Kompleksitas Pengembangan: Mengembangkan integrasi antara ESP32-CAM, *Firebase Firestore*, dan *Telegram Bot* memerlukan pemahaman yang mendalam tentang masing-masing *platform* serta kemampuan untuk mengelola API dan koneksi. Ini dapat meningkatkan kompleksitas pengembangan dan memerlukan waktu tambahan untuk konfigurasi dan debugging.
2. Keterbatasan Fitur *Telegram Bot*: Meskipun *Telegram Bot* menawarkan antarmuka yang kuat untuk pengiriman pesan dan *file*, ada beberapa batasan terkait dengan fungsi-fungsi tertentu yang mungkin tidak sepenuhnya memenuhi kebutuhan aplikasi yang lebih kompleks. Hal ini perlu dipertimbangkan dalam merancang fungsionalitas dan respons sistem yang diinginkan.
3. Aplikasi android yang dikembangkan untuk menampilkan pada halaman utama belum menampilkan gambar sehingga saat ingin memeriksa *log* harus membuka *telegram* untuk melihat

**4.2.3 Penjelasan Pergantian *Blynk* ke *Firebase Firestore* dan *Telegram Bot***:

Pergantian dari *Blynk* ke *Firebase Firestore* dan *Telegram Bot* terutama didorong oleh pertimbangan biaya dan fleksibilitas dalam pengembangan aplikasi. Blynk, meskipun menawarkan fitur yang kuat untuk pengendalian perangkat IoT dan visualisasi data secara *real-time*, memerlukan biaya berlangganan untuk mengakses fitur-fitur lanjutan yang mungkin tidak diperlukan pada tahap awal pengembangan atau uji coba proyek.

*Firebase Firestore*, di sisi lain, menyediakan basis data yang dapat diakses secara real-time tanpa biaya berlangganan tambahan yang signifikan. Ini membuatnya lebih hemat biaya untuk penggunaan proyek eksperimental atau skala kecil yang masih dalam tahap pengembangan. *Firebase* juga memiliki integrasi yang baik dengan platform lain dan menyediakan solusi yang skalabel untuk penyimpanan dan sinkronisasi data.

*Telegram Bot* dipilih sebagai pengganti antarmuka *Blynk* karena menyediakan solusi yang sederhana dan efisien untuk interaksi pengguna dengan perangkat IoT. *Telegram Bot API* memungkinkan untuk mengirim dan menerima pesan serta file, termasuk foto yang diambil oleh kamera ESP32-CAM, dengan responsivitas yang baik. Penggunaan *Telegram Bot* juga menghilangkan kebutuhan akan aplikasi tambahan pada perangkat pengguna, karena aplikasi *Telegram* sudah umum digunakan dan tersedia di berbagai *platform.*

Dengan demikian, kombinasi *Firebase Firestore* dan *Telegram Bot* tidak hanya memberikan solusi yang lebih terjangkau secara finansial untuk proyek ini, tetapi juga memungkinkan pengembang untuk lebih fleksibel dalam mengatur interaksi antara perangkat keras dan pengguna. Meskipun ada tantangan dalam ketergantungan pada koneksi internet dan kompleksitas pengembangan, kelebihan yang ditawarkan oleh sistem ini jauh lebih menguntungkan dalam mengoptimalkan pengalaman pengguna dan efisiensi operasional pada tingkat aplikasi IoT.

**BAB V**

**KESIMPULAN DAN SARAN**

###### Kesimpulan

Adapun kelebihan sistem usulan yang dirancang oleh penulis antara lain sebagai berikut :

1. Pemantauan secara *real-time* karena Kamera ESP32CAM dapat mengirim foto secara *real-time* ke *Telegram*, memungkinkan pengguna untuk memantau area tertentu secara langsung dari jarak jauh
2. Sensor PIR digunakan sebagai pendeteksi Gerakan, sehingga dapat mengirimkan informasi kepada pengguna untuk memilih Langkah dalam mengambil foto.
3. Efisiensi energi pada sistem cenderung lebih efisien dalam penggunaan daya karena ESP32CAM bisa diatur dalam mode tidur dan hanya aktif saat sensor PIR mendeteksi Gerakan.
4. Kemudahan dalam megintegrasikan dengan Telegram memungkinkan pengguna menerima notifikasi dan foto langsung di aplikasi yang sudah umum digunakan, tanpa memerlukan setup atau aplikasi tambahan yang rumit.
5. Hanya memerlukan koneksi wifi. ESP32CAM memiliki kemampuan koneksi Wi-Fi yang memungkinkan perangkat ini terhubung ke jaringan dan mengirimkan data tanpa memerlukan kabel tambahan

###### Saran

Selain dari kelebihan sistem yang dirancang juga memiliki kekurangan antara

lain sebagai berikut:

1. Integrasi dengan Sistem Alarm: Integrasikan sistem ini dengan alarm berbasis suara atau notifikasi tambahan untuk memberikan peringatan langsung di tempat ketika gerakan terdeteksi.
2. Deteksi Wajah: Implementasikan algoritma deteksi wajah untuk mengidentifikasi manusia dan mengurangi notifikasi yang disebabkan oleh gerakan tidak relevan.
3. Penyimpanan *Cloud*: Tambahkan opsi untuk menyimpan foto dan video langsung ke layanan penyimpanan *cloud* seperti *Google Drive* atau *Dropbox*, sebagai backup jika penyimpanan lokal penuh atau rusak.
4. Peningkatan Antarmuka Pengguna: Buat dashboard berbasis web atau aplikasi *mobile* untuk memudahkan pengguna dalam mengatur dan memantau sistem secara lebih intuitif.
5. Pelatihan Model AI:Latih model AI untuk mengenali jenis-jenis gerakan yang berbeda, seperti pergerakan manusia vs. binatang, sehingga hanya gerakan relevan yang memicu pengiriman notifikasi.

**DAFTAR PUSTAKA**

[1] S. Ding, A. Tukker and H. Ward, “Opportunities and risks of internet of things (IoT) technologies for circular business models: A literature review,” *Journal of Environmental Management,* vol. 336, pp. 1-13, 15 June 2023.

[2] R. R. Rohmansyah and H. Nurwasito, “Pengembangan Aplikasi Mobile untuk Sistem Keamanan Kantor Menggunakan NFC (Near Field Communication) dan Wi-Fi,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* vol. 2, no. 1, pp. 81-90, Januari 2018.

[3] P. B. Statistik, “https://www.bps.go.id,” 11 2022. [Online]. Available: https://www.bps.go.id/publication/2022/11/30/4022d3351bf3a05aa6198065/statistik-kriminal-2022.html. [Accessed 4 7 2023].

[4] N. D. Aulia and D. N. Bogi A. K, S.T., MSEE., “PERANCANGAN DAN IMPLEMENTASI SISTEM KEAMANAN RUMAH,” *e-Proceeding of Engineering,* vol. 6, no. 2, pp. 4488-4494, 2019.

[5] M. R. Ferdiansyah, A. Wibowo and S. , “PERANCANGAN ROBOT MONITORING KEAMANAN KANTOR MENGGUNAKAN MIKROKONTROLER ESP32CAM BERBASIS INTERNET OF THINGS,” *Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI),* vol. 2, no. 1, pp. 399-406, 1 April 2023.

[6] S. Erpanji, R. Munadi and S. N. Hertian, “PERANCANGAN DAN IMPLEMENTASI SISTEM KEAMANAN KANTOR MENGGUNAKAN CCTV DENGAN MEDIA APLIKASI ANDROID,” *e-Proceeding of Engineering,* vol. 8, no. 5, pp. 5325-5334, Oktober 2021.

[7] M. I. Ali, S. A. Wibowo and A. Panji, “KEAMANAN BRANKAS MENGGUNAKAN E-KTP DAN NOTIFIKASI VIA TELEGRAM BERBASIS IOT (INTERNET OF THINGS),” *JATI (Jurnal Mahasiswa Teknik Informatika),* vol. 5, no. 2, pp. 589-595, September 2021.

[8] M. I. Ali, S. A. Wibowo and A. Panji, “Perancangan Sistem Controller Lighting and air conditioner di Unisla Dengan Konsep Internet Of Things (Iot) Berbasis Web,” *Generation Journal,* vol. 6, no. 1, pp. 37-44, 2022.

[9] O. Pribadi and J. , “Perancangan Simulasi Sistem Otentikasi Pengguna Menggunakan Perangkat Radio Frequency Identification (RFID) Dengan Konsep Internet Of Things(IoT),” *JURNAL TIMES,* vol. VIII, no. 2, pp. 19-23, 2019.

[10] R. Genaldo, T. Septyawan, A. Surahman and P. , “SISTEM KEAMANAN PADA RUANGAN PRIBADI MENGGUNAKAN MIKROKONTROLER ARDUINO DAN SMS GATEWAY,” *JTIKOM,* vol. 1, no. 2, pp. 46-52, 2022.

[11] A. Setiawan and A. I. Purnamasari, “Pengembangan Smart Home Dengan Microcontrollers ESP32 Dan MC-38 Door Magnetic Switch Sensor Berbasis Internet of Things (IoT) Untuk Meningkatkan Deteksi Dini Keamanan Perumahan,” *JURNAL RESTI,* vol. 3, no. 3, pp. 451-457, 2019.

[12] A. Sahara, H. F. Oktafiani and M. Fari, “SISTEM OTOMASI PENGHITUNG PENGUNJUNG STADION BASKET MENGGUNAKAN SENSOR PIR BERBASIS ARDUINO UNO,” *PETROGAS,* vol. 5, no. 2, October 2022.

[13] A. Ramschie, J. Makal, R. Katuuk and V. Ponggawa, “Pemanfaatan ESP32 Pada Sistem Keamanan Rumah Tinggal Berbasis IoT,” *IRWNS,* pp. 175-181, 2021.

[14] W. Raditya, S. A. Surahman and A. Budi, “PENERAPAN SISTEM KEAMANAN GERBANG RUMAH BERBASIS TELEGRAM MENGGUNAKAN ESP8266,” *JTIKOM,* vol. 3, no. 2, pp. 93-103, 2022.

[15] Kemdikbud. [Online]. Available: https://kbbi.kemdikbud.go.id/. [Accessed 2 2024].

**Listing Program**

**Program membuat koneksi Arduino ke firebase dan telegram**

#include <Arduino.h>

#include <WiFi.h>

#include <WiFiClientSecure.h>

#include "soc/soc.h"

#include "soc/rtc\_cntl\_reg.h"

#include "esp\_camera.h"

#include <UniversalTelegramBot.h>

#include <ArduinoJson.h>

#include <HTTPClient.h> // Tambahkan library HTTPClient

const char\* ssid = "Vincent";

const char\* password = "vin91382";

// Initialize Telegram BOT

String BOTtoken = "7341127581:AAHgPjEdA36kVOHnZ3gpwWJSfsVRapqp8J0";

String CHAT\_ID = "1812377475";

bool sendPhoto = false;

WiFiClientSecure clientTCP;

UniversalTelegramBot bot(BOTtoken, clientTCP);

#define FLASH\_LED\_PIN 4

#define PIR\_PIN 13 // Deklarasikan pin untuk PIR sensor

bool flashState = LOW;

// Checks for new messages every 1 second.

int botRequestDelay = 1000;

unsigned long lastTimeBotRan;

// CAMERA\_MODEL\_AI\_THINKER

#define PWDN\_GPIO\_NUM 32

#define RESET\_GPIO\_NUM -1

#define XCLK\_GPIO\_NUM 0

#define SIOD\_GPIO\_NUM 26

#define SIOC\_GPIO\_NUM 27

#define Y9\_GPIO\_NUM 35

#define Y8\_GPIO\_NUM 34

#define Y7\_GPIO\_NUM 39

#define Y6\_GPIO\_NUM 36

#define Y5\_GPIO\_NUM 21

#define Y4\_GPIO\_NUM 19

#define Y3\_GPIO\_NUM 18

#define Y2\_GPIO\_NUM 5

#define VSYNC\_GPIO\_NUM 25

#define HREF\_GPIO\_NUM 23

#define PCLK\_GPIO\_NUM 22

void configInitCamera() {

camera\_config\_t config;

config.ledc\_channel = LEDC\_CHANNEL\_0;

config.ledc\_timer = LEDC\_TIMER\_0;

config.pin\_d0 = Y2\_GPIO\_NUM;

config.pin\_d1 = Y3\_GPIO\_NUM;

config.pin\_d2 = Y4\_GPIO\_NUM;

config.pin\_d3 = Y5\_GPIO\_NUM;

config.pin\_d4 = Y6\_GPIO\_NUM;

config.pin\_d5 = Y7\_GPIO\_NUM;

config.pin\_d6 = Y8\_GPIO\_NUM;

config.pin\_d7 = Y9\_GPIO\_NUM;

config.pin\_xclk = XCLK\_GPIO\_NUM;

config.pin\_pclk = PCLK\_GPIO\_NUM;

config.pin\_vsync = VSYNC\_GPIO\_NUM;

config.pin\_href = HREF\_GPIO\_NUM;

config.pin\_sscb\_sda = SIOD\_GPIO\_NUM;

config.pin\_sscb\_scl = SIOC\_GPIO\_NUM;

config.pin\_pwdn = PWDN\_GPIO\_NUM;

config.pin\_reset = RESET\_GPIO\_NUM;

config.xclk\_freq\_hz = 20000000;

config.pixel\_format = PIXFORMAT\_JPEG;

config.grab\_mode = CAMERA\_GRAB\_LATEST;

if (psramFound()) {

config.frame\_size = FRAMESIZE\_UXGA;

config.jpeg\_quality = 10;

config.fb\_count = 1;

} else {

config.frame\_size = FRAMESIZE\_SVGA;

config.jpeg\_quality = 12;

config.fb\_count = 1;

}

esp\_err\_t err = esp\_camera\_init(&config);

if (err != ESP\_OK) {

Serial.printf("Camera init failed with error 0x%x", err);

delay(1000);

ESP.restart();

}

}

void handleNewMessages(int numNewMessages) {

Serial.print("Handle New Messages: ");

Serial.println(numNewMessages);

for (int i = 0; i < numNewMessages; i++) {

String chat\_id = String(bot.messages[i].chat\_id);

if (chat\_id != CHAT\_ID) {

bot.sendMessage(chat\_id, "Unauthorized user", "");

continue;

}

String text = bot.messages[i].text;

Serial.println(text);

String from\_name = bot.messages[i].from\_name;

if (text == "/start") {

String welcome = "Welcome , " + from\_name + "\n";

welcome += "Use the following commands to interact with the ESP32-CAM \n";

welcome += "/photo : takes a new photo\n";

welcome += "/flash : toggles flash LED \n";

bot.sendMessage(CHAT\_ID, welcome, "");

}

if (text == "/flash") {

useFlash();

}

if (text == "/photo") {

sendPhoto = true;

Serial.println("New photo request");

sendHttpPostRequest();

Serial.println("sendHttpPost true");

}

}

}

String sendPhotoTelegram() {

const char\* myDomain = "api.telegram.org";

String getAll = "";

String getBody = "";

camera\_fb\_t \* fb = NULL;

fb = esp\_camera\_fb\_get();

esp\_camera\_fb\_return(fb);

fb = NULL;

fb = esp\_camera\_fb\_get();

if (!fb) {

Serial.println("Camera capture failed");

delay(1000);

ESP.restart();

return "Camera capture failed";

}

Serial.println("Connect to " + String(myDomain));

if (clientTCP.connect(myDomain, 443)) {

Serial.println("Connection successful");

String head = "--RandomNerdTutorials\r\nContent-Disposition: form-data; name=\"chat\_id\"; \r\n\r\n" + CHAT\_ID + "\r\n--RandomNerdTutorials\r\nContent-Disposition: form-data; name=\"photo\"; filename=\"esp32-cam.jpg\"\r\nContent-Type: image/jpeg\r\n\r\n";

String tail = "\r\n--RandomNerdTutorials--\r\n";

size\_t imageLen = fb->len;

size\_t extraLen = head.length() + tail.length();

size\_t totalLen = imageLen + extraLen;

clientTCP.println("POST /bot" + BOTtoken + "/sendPhoto HTTP/1.1");

clientTCP.println("Host: " + String(myDomain));

clientTCP.println("Content-Length: " + String(totalLen));

clientTCP.println("Content-Type: multipart/form-data; boundary=RandomNerdTutorials");

clientTCP.println();

clientTCP.print(head);

uint8\_t \*fbBuf = fb->buf;

size\_t fbLen = fb->len;

for (size\_t n = 0; n < fbLen; n = n + 1024) {

if (n + 1024 < fbLen) {

clientTCP.write(fbBuf, 1024);

fbBuf += 1024;

} else if (fbLen % 1024 > 0) {

size\_t remainder = fbLen % 1024;

clientTCP.write(fbBuf, remainder);

}

}

clientTCP.print(tail);

esp\_camera\_fb\_return(fb);

int waitTime = 10000;

long startTimer = millis();

boolean state = false;

while ((startTimer + waitTime) > millis()) {

Serial.print(".");

delay(100);

while (clientTCP.available()) {

char c = clientTCP.read();

if (state == true) getBody += String(c);

if (c == '\n') {

if (getAll.length() == 0) state = true;

getAll = "";

} else if (c != '\r')

getAll += String(c);

startTimer = millis();

}

if (getBody.length() > 0) break;

}

clientTCP.stop();

Serial.println(getBody);

} else {

getBody = "Connected to api.telegram.org failed.";

Serial.println("Connected to api.telegram.org failed.");

}

return getBody;

}

// Fungsi untuk melakukan permintaan HTTP POST

void sendHttpPostRequest() {

String url = "https://5c6d54ea-55b7-4b58-a9ff-fd87cf4f3d61-00-ylrivcx8auk0.sisko.replit.dev/add";

HTTPClient http;

String response;

http.begin(url);

int httpResponseCode = http.GET();

if (httpResponseCode > 0) {

response = http.getString();

Serial.println("HTTP Response code: " + String(httpResponseCode));

Serial.println("Response: " + response);

} else {

Serial.println("Error on HTTP request");

}

http.end();

}

void setup() {

WRITE\_PERI\_REG(RTC\_CNTL\_BROWN\_OUT\_REG, 0);

Serial.begin(115200);

pinMode(FLASH\_LED\_PIN, OUTPUT);

digitalWrite(FLASH\_LED\_PIN, flashState);

pinMode(PIR\_PIN, INPUT); // Inisialisasi pin PIR sebagai input

configInitCamera();

WiFi.mode(WIFI\_STA);

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

clientTCP.setCACert(TELEGRAM\_CERTIFICATE\_ROOT);

while (WiFi.status() != WL\_CONNECTED) {

Serial.print(".");

delay(500);

}

Serial.println();

Serial.print("ESP32-CAM IP Address: ");

Serial.println(WiFi.localIP());

}

void loop() {

if (digitalRead(PIR\_PIN) == HIGH) { // Jika PIR mendeteksi gerakan

Serial.println("Motion detected by PIR sensor");

sendPhoto = true; // Setel flag untuk mengirim foto

}

if (sendPhoto) {

Serial.println("Preparing photo");

useFlash();

sendHttpPostRequest();

sendPhotoTelegram();

sendPhoto = false; // Setel flag kembali ke false setelah mengirim foto

useFlash();

}

if (millis() > lastTimeBotRan + botRequestDelay) {

int numNewMessages = bot.getUpdates(bot.last\_message\_received + 1);

while (numNewMessages) {

Serial.println("got response");

handleNewMessages(numNewMessages);

numNewMessages = bot.getUpdates(bot.last\_message\_received + 1);

}

lastTimeBotRan = millis();

}

}

void useFlash() {

flashState = !flashState;

digitalWrite(FLASH\_LED\_PIN, flashState);

Serial.println("Change flash LED state");

}

**RIWAYAT HIDUP**

**Nama** : **VINCENT**

Tempat/Tanggal Lahir : Medan,28 April 2002

Jenis Kelamin : Laki-Laki

Agama : Buddha

Alamat : Jl. AMD Komplek Dewi Indah D5

Email : vincpr71@gmail,com

**Pendidikan Formal** :

2008 – 2014 SD Dr. Wahidin Sudirohusodo

2014 – 2017 SMP Dr. Wahidin Sudirohusodo

2017 – 2020 SMA Dr. Wahidin Sudirohusodo

2020 – 2024 S1 Teknik Informatika STMIK TIME Medan

**Pengalaman Kerja** :

Oktober 2020 – November 2021 Admin Penjualan di PT Platina Mulia Abadi

Februari 2022 – Mei 2024 Staff IT Development di PT Industri Pembungkus Internasional

Juni 2024 – Sekarang Programmer Officer di PT Musim Mas

Medan,Juni 2024

Vincent