



Sistem Keamanan Menggunakan Alarm Berbasis Internet Of Things (IOT)

Didik Trisianto¹, Muhammad Julian Ilham Rizandy²

¹Ilmu Komputer, Universitas Narotama, Surabaya, Indonesia, didik.trisianto@narotama.ac.id

²Ilmu Komputer, Universitas Narotama, Surabaya, Indonesia,

STATUS ARTIKEL

Dikirim 07 Juli 2023

Direvisi 14 Juli 2023

Diterima 21 Juli 2023

Kata Kunci:

ESP32-CAM, Internet of Things, MIT

App Inventor, sensor PIR, sistem

keamanan

ABSTRAK

Kejahatan sering terjadi di daerah yang padat penduduknya. Kompleks perumahan yang tidak memiliki pengawasan langsung dan rutin terhadap sekitarnya dapat membuat penduduk khawatir akan kegiatan kriminal. Di era Internet of Things (IoT), kita dapat memanfaatkan teknologi ini untuk merancang dan membangun sistem keamanan yang canggih. Dengan menggabungkan sistem yang dapat memberikan peringatan segera kepada penghuni dalam kasus penyusupan, kita dapat meningkatkan langkah-langkah keamanan. Mengembangkan solusi keamanan berbasis Internet of Things melibatkan penggunaan modul ESP32-CAM sebagai kontroler utama, bersama dengan sensor PIR untuk mendeteksi gerakan manusia. Selain itu, MIT App Inventor dapat digunakan sebagai antarmuka pengguna, memungkinkan pengguna untuk dengan mudah memasang sistem ini di rumah atau gedung mereka dan mengaksesnya melalui aplikasi yang mudah digunakan. Dalam hal keamanan, sistem ini memberikan respons cepat terhadap aktivitas mencurigakan, memastikan keselamatan pengguna dan melindungi aset mereka. Dengan menggunakan Firebase sebagai database, data keamanan dapat disimpan dengan aman dan dapat diakses dengan mudah oleh pengguna.

1. PENDAHULUAN

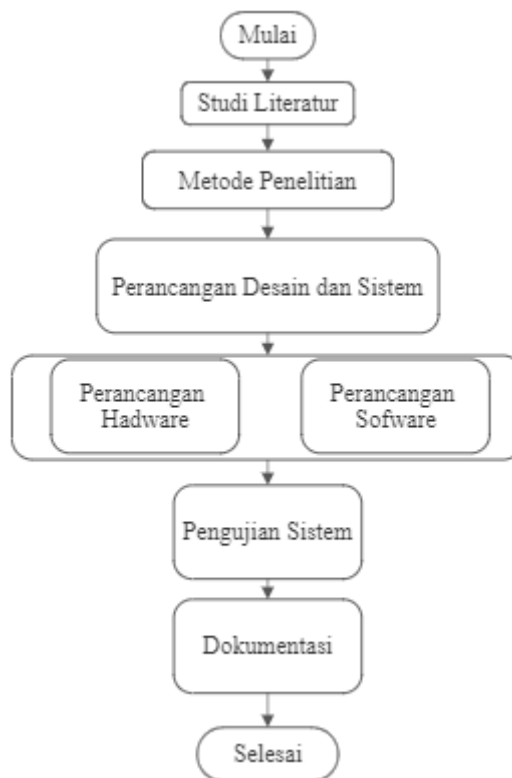
Terjadinya banyak kasus pencurian di Indonesia tentu saja membuat masyarakat geram dikarenakan kerugian yang disebabkan begitu banyak dan pelaku pencurian yang tidak pernah jera. Mulai dari, pencurian sepeda motor, mobil, perhiasan, maupun barang berharga lainnya yang menyebabkan kerugian puluhan hingga ratusan juta rupiah. Keamanan adalah salah satu aspek yang sangat penting dan diperlukan dalam kehidupan sekarang dan yang akan datang, hal ini dikarenakan faktor privasi yang sangat mempengaruhi dalam bidang keamanan. Namun, disisi lain secara tidak langsung para pelaku pencurian ini menjadi salah satu penyebab berkembangnya sistem keamanan sehingga tercipta sistem keamanan yang baru dan lebih maju dari sebelumnya untuk mencegah para pelaku melakukan tindakan pencurian.

Munculnya Internet of Things (IoT) telah mendorong terciptanya solusi inovatif di berbagai bidang terutama pada sistem keamanan. Sistem keamanan tradisional menjadi tidak memadai karena pencuri dan pembobol menemukan cara untuk melewatinya (Ali, Ali, & Badawy, October 2015). Sistem keamanan berbasis IoT menawarkan keamanan yang lebih baik dengan memberikan peringatan secara real-time dan memungkinkan pemantauan jarak jauh. Penelitian ini diharapkan akan mengembangkan alarm keamanan yang dapat mendeteksi gerakan menggunakan sensor yang sudah dirancang khusus. Kemudian, sistem keamanan akan

berkomunikasi dengan alat pemantau pusat melalui akses internet. Sistem pengawas ini akan memberikan sinyal peringatan ketika terdeteksi adanya penyusup, dan memungkinkan pemantauan jarak jauh dari lokasi tersebut. Alat yang dirancang juga akan mengirimkan foto/video secara langsung ke smartphone pemilik sistem keamanan.

2. METODE

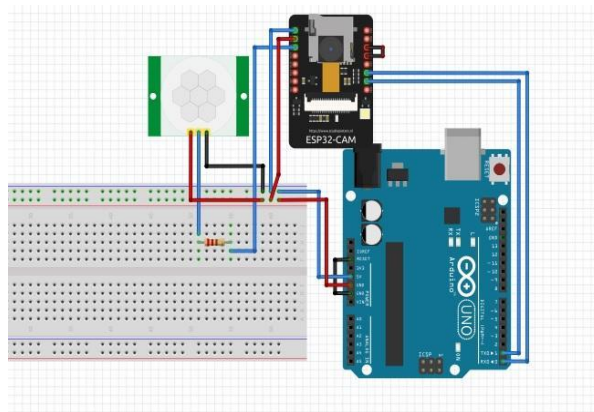
Riset diharapkan mampu menghasilkan output yang tepat terarah. Penelitian dilakukan melalui tahapan yang panjang dan keterkaitan antar sumber informasi yang ada. Riset yang sistematis harus dilakukan agar hasil penelitian dapat dipertanggungjawabkan dan dapat menjadi sumber literatur untuk perkembangan penelitian selanjutnya. Agar riset memiliki panduan dalam melakukan proses penelitian, maka diperlukan desain penelitian. Metode penelitian yang digunakan penulis yaitu, Penelitian literatur, analisis masalah, perencanaan dan perancangan sistem, implementasi sistem, pengujian sistem, evaluasi dan dokumentasi.



Gambar 1 Diagram Penulisan

2.1 Perancangan Hardware

Hal Awal yang perlu dilakukan ialah menyusun perancangan sistem alat detektor keamanan berbasis internet of things yang memakai mikrokontroler ESP32-Cam, kemudian menggunakan sensor PIR yang dipasang akan membaca data yang berupa pergerakan apabila terdeteksi suatu pergerakan, setelah mikrokontroler mendapat sebuah input dari sensor tersebut maka data yang diolah akan dikirim pada firebase untuk ditampilkan hasilnya di aplikasi apakah ada penyusup atau aman, dibawah ini ada gambar rancang bangun sistem alat detector keamanan berbasis internet of things.



Gambar 2 Rangkaian Pemasangan Hadware

Gambar diatas merupakan rangkaian pemasangan hardware yang terdiri dari beberapa komponen yang dihubungkan satu sama lain supaya sistem dapat berfungsi dengan baik, terdapat beberapa komponen yang terlihat, dan berikut adalah penjelasan tentang pemasangan hardware tersebut ;

1) Arduino Uno

Arduino Uno berfungsi sebagai penghubung antara ESP32-Cam dengan Laptop, dikarenakan Esp32 cam tidak memiliki port Usb sehingga memerlukan Arduino Uno sebagai perantara sebagai Source Code dan catu daya. Port yang dihubungkan antara Arduino Uno dengan Esp32-Cam ialah pin RX & TX dengan pin RX & TX pada ESP32 Cam sebagai penghubung, pin Reset Arduino dihubungkan ke pin GND.

2) ESP32-Cam

ESP32-Cam adalah komponen utama dalam rangkaian ini, ESP32- Cam memiliki kamera yang akan terhubung dengan internet sehingga dapat menampilkan hasil rekaman video secara live. ESP32-Cam juga akan mengendalikan sensor PIR untuk mendeteksi pergerakan suhu tubuh melalui pin IO12 yang terhubung pada resistor 220V.

3) Sensor PIR

Sensor PIR merupakan sensor utama untuk mendeteksi pergerakan suhu tubuh yang dihubungkan ke ESP32-Cam melalui port IO12.

2.2 Perancangan Software

Perancangan perangkat lunak berupa antarmuka aplikasi yang akan digunakan untuk memonitoring ketika terjadi pergerakan pada area sensor pir, dimana pada perancangan ini meliputi pembuatan instalasi software IDE, pembuatan database dan perancangan aplikasi pada platform MIT App Inventor dan smartphone sebagai media monitoring.

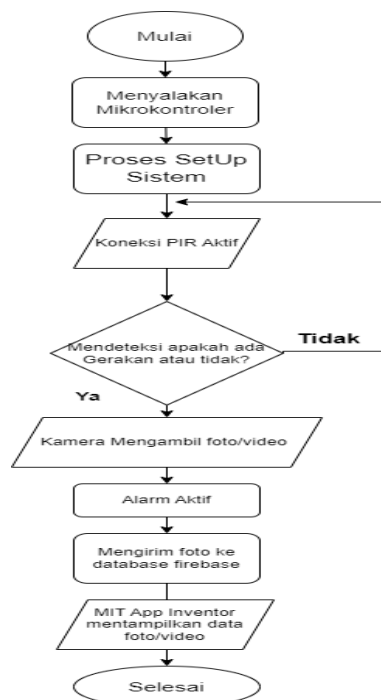
1) Pembuatan Database

Firestore merupakan salah satu layanan dari Google yang berfungsi untuk memudahkan para developer mengembangkan sebuah aplikasi. Firestore juga merupakan

salah satu cara yang disuguhkan oleh Google bertujuan untuk memudahkan para pengguna. Cara pembuatan database Firebase adalah dengan membuka website terlebih dahulu dan setelah itu login dan mengikuti langkah-langkah yang tersedia, kemudian barulah dapat menggunakan fasilitas tersebut.

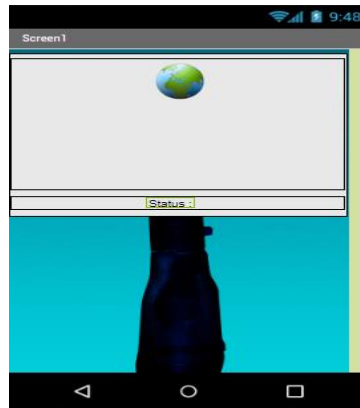
2) MIT App Inventor

MIT App Inventor merupakan sebuah software open source yang digunakan untuk membangun beberapa program aplikasi pada sistem operasi Android berbasis internet. Pembuatan aplikasi menggunakan MIT App Inventor ini akan menjadi penghubung antara User dengan sistem yang akan dibuat. Adapun untuk lebih lengkapnya akan ditampilkan flowchart dari perangkat lunak sistem alarm keamanan.



Gambar 3 Diagram Alur Aplikasi

Gambar tersebut merupakan gambar yang menunjukkan bahwa bagaimana cara kerja alur sistem perangkat software, dari dimulainya perangkat berjalan dan saat dinyalakan alat akan otomatis melakukan searching untuk mencari sambungan internet yang aktif, saat sambungan sudah aktif maka sensor PIR akan otomatis aktif, kemudian selanjutnya jika sensor PIR mendeteksi adanya gerakan maka alarm akan berbunyi, ditahapan ini terdapat dua opsi apabila tidak terdeteksi adanya gerakan maka audio alarm tidak akan menyala dan kembali ke jalur sambungan yang aktif, apabila sensor PIR mendeteksi adanya pergerakan maka alarm akan berbunyi dan sistem akan mengirimkan foto serta laporan ke cloudbase Firebase aplikasi user yang sudah diinstall di smartphone pengguna, pengguna bisa melihat laporan yang terdeteksi oleh sensor PIR.

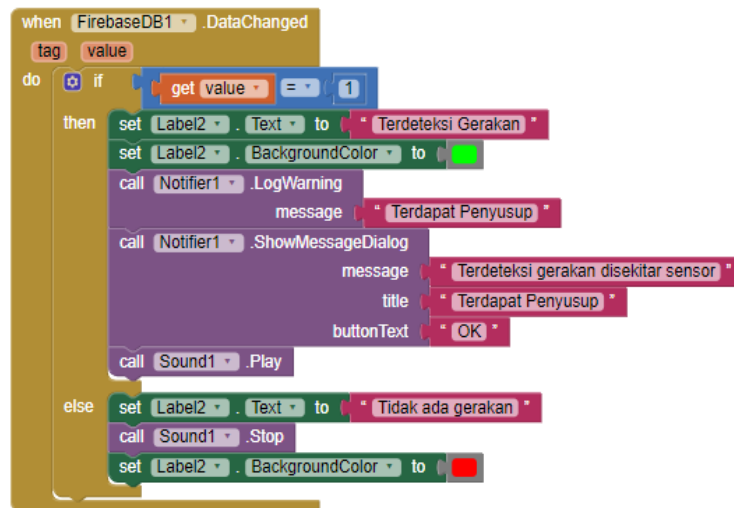


Gambar 4 User awal Interface Monitoring

Gambar diatas merupakan tampilan User Interface yang nantinya akan digunakan untuk memonitoring keamanan sekitar dan memberikan informasi kepada pengguna

3) Perancangan Block Sistem Android

Pada rancangan membuat blok aplikasi ini juga menggunakan Web MIT APP Inventor yang proses pembuatan blok ini adalah memasukkan program perintah, dimana nantinya hasil dari sensornya akan ditampilkan melalui aplikasi UI yang telah dibuat.



Gambar 5 Perancangan Block

Gambar di atas merupakan tampilan desain block diagram yang digunakan untuk memonitoring keamanan sekitar dan memberikan informasi kepada pengguna.

3. HASIL DAN PEMBAHASAN

Pada bab ini pengujian akan dilakukan untuk mengetahui apakah hardware dan software yang dirancang sudah berfungsi dengan baik. Pengujian sistem yang dilakukan meliputi pengujian perangkat keras atau alat, pengujian perangkat lunak sistem, dan pengujian tertulis secara keseluruhan sistem.

3.1 Pengujian Hardware

Eksperimen perangkat keras ini harus dilakukan untuk mengetahui apakah komponen yang digunakan berfungsi dengan baik. Dengan menghubungkan semua komponen dan mengirimkan kode program dari software Arduino IDE ke mikrokontroler ESP-32cam.

1) Pengujian Mikrokontroler ESP 32-cam

Untuk memastikan keakuratan dan fungsi mikrokontroler Esp32-cam terkoneksi ke jaringan internet WiFi. Pengujian mikrokontroler ini diharapkan dapat mengurangi kesalahan teknis karena adanya proses pengujian sistem secara keseluruhan. Dibawah ini merupakan coding Arduino IDE pada mikrokontroler ESP32-cam

```

FirebaseCam$  app_httpd.cpp  camera_index.h  camera_pins.h
#include "esp_camera.h"
#include <WiFi.h>
#include <FirebaseESP32.h>

#define CAMERA_MODEL_AI_THINKER // Has PSRAM

#include "camera_pins.h"

#define FIREBASE_HOST "riset-andy-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "Dvfv8QWiOfFIE5T5yxQXuAhKcEclJYKsfm2qGOS1"
#define WIFI_SSID "OPPO Rz13"
#define WIFI_PASSWORD "andyriz12345"

FirebaseData fbdo;
int count = 0;
int isObstaclePin = 12; // Pin input
int isObstacle = HIGH;
    
```

Gambar 6 Source Code Arduino IDE

Dalam kode program menggunakan library “esp_camera.h” untuk menjalankan code program ESP32-Cam, lalu juga menggunakan library <WiFi.h> untuk menghubungkan wifi ESP32-Cam dan menggunakan library <FirebaseESP32.h> agar dapat menerima dan mengirim data pada database Firebase. Selanjutnya deklarasi variabel Database Firebase, URL Database Firebase, Firebase kode autentikasi, SSID dan Password Wifi, serta pin yang terhubung pada sensor PIR

```

void startCameraServer();

void setup() {
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  Serial.println();

  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;

```

Gambar 7 Deklarasi pin camera pada ESP32-Cam

Dengan menggunakan library “camera_pins” pada Gambar di atas sehingga perlu mendeklarasikan pin camera pada ESP32-Cam.

```

if(psramFound()){
  config.frame_size = FRAMESIZE_UXGA;
  config.jpeg_quality = 10;
  config.fb_count = 2;
} else {
  config.frame_size = FRAMESIZE_SVGA;
  config.jpeg_quality = 12;
  config.fb_count = 1;
}

#ifdef CAMERA_MODEL_ESP_EYE
  pinMode(13, INPUT_PULLUP);
  pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
  Serial.printf("Camera init failed with error 0x%x", err);
  return;
}

```

```

sensor_t * s = esp_camera_sensor_get();
// initial sensors are flipped vertically and colors are a bit saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1); // flip it back
    s->set_brightness(s, 1); // up the brightness just a bit
    s->set_saturation(s, -2); // lower the saturation
}
// drop down frame size for higher initial frame rate
s->set_framesize(s, FRAMESIZE_QVGA);

#ifdef CAMERA_MODEL_MSSTACK_WIDE || defined(CAMERA_MODEL_MSSTACK_ESP32CAM)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#endif

```

Gambar 8 Source code Konfigurasi kamera

Kamera pada ESP32-Cam akan dikonfigurasi terlebih dahulu untuk mengecek ketersediaan PSRAM, pengecekan modul kamera, pengaturan sensor serta pengaturan tambahan untuk model kamera tertentu.

```

WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

startCameraServer();

Serial.print("Camera Ready! Use 'http://");
Serial.print(WiFi.localIP());
Serial.println("' to connect");

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

```

Gambar 9 Source code pengaturan Wifi

Setelah itu ESP32-Cam akan menghubungkan terhadap variabel SSID dan Password Wifi yang telah diinputkan pada Gambar di atas lalu ESP32-Cam akan menghubungkan dengan URL Firebase database dan kode autentikasi Firebase.

2) Pengujian Sensor PIR (Passive Infrared Sensor)

Pengujian pada sensor Motion PIR HCSR501 akan membaca pergerakan yang dilakukan oleh manusia atau hewan melalui suhu tubuh yang kemudian data tersebut akan dikirimkan oleh ESP32-Cam menuju Firebase Database, jika terdapat pergerakan makhluk hidup sensor akan mengirimkan data '1' dan jika tidak akan mengirimkan '0'.


```

isObstacle = digitalRead(isObstaclePin);
if (isObstacle == LOW) {
Serial.println("Depan hanya kosong");
count = 0;
Serial.printf("Data \n", Firebase.setInt(fbdo, "/Data/int", count));
  Serial.printf("Data saat ini : %s\n", Firebase.getInt(fbdo, "/Data/int") ? String(fbdo.to<int>()).c_str() : fbdo.errorReason().c_str());
  delay(1000);
  //digitalWrite(LED_BUILTIN, HIGH);
} else {
Serial.println("Terdeteksi di depan!");
count = 1;
Serial.printf("Data \n", Firebase.setInt(fbdo, "/Data/int", count));
  Serial.printf("Data saat ini : %s\n", Firebase.getInt(fbdo, "/Data/int") ? String(fbdo.to<int>()).c_str() : fbdo.errorReason().c_str());
  delay(1000);
// digitalWrite(LED_BUILTIN, LOW);
}
delay(200);
  delay(10000);
}

```

Gambar 10 Source code PIR

Pada pengujian program diatas dapat diketahui bahwa sensor Motion PIR HCSR501 dapat mendeteksi pada jarak optimal 7 Cm, dengan mengatur sensitivitas sensor dapat dilakukan dengan memutar potensiometer yang ada pada modul sensor seperti gambar dbawah ini



Gambar 11 Tombol Potensioner Module Pada PIR

Selanjutnya mengetahui hasil dari kecepatan pengujian sistem pada sensor PIR dibuat dalam bentuk tabel dibawah sebagai berikut ;

Tabel 1 Pengujian Jarak Sensor PIR

Jarak	Sensor PIR	Durasi Data terima
1 m	Detected	5 Second
2 m	Detected	6 second
3 m	Detected	5 second
4 m	Detected	6 second
5 m	detected inaccurately	6 second
6 m	detected inaccurately	6 second
7 m	detected inaccurately	6 second

3) Hasil Pengujian Hardware

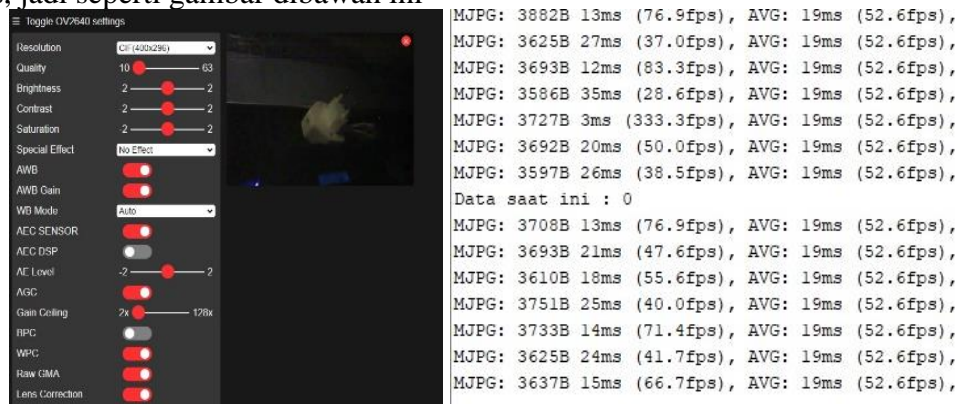
Hasil uji pertama apakah ESP32-cam berhasil tersambung atau tidak dengan hotspot Wifi yang digunakan, jika berhasil maka akan terlihat seperti ini.

```

.....
WiFi connected
Starting web server on port: '80'
Starting stream server on port: '81'
Camera Ready! Use 'http://192.168.43.250' to connect
    
```

Gambar 12 Esp32-cam terhubung dengan Wifi

Selanjutnya untuk mengecek apakah kamera pada esp32-cam berfungsi dengan menggunakan link yang tercatum di code diatas lalu buka web dan mengcopy link diatas, jadi seperti gambar dibawah ini



Gambar 13 Webcam

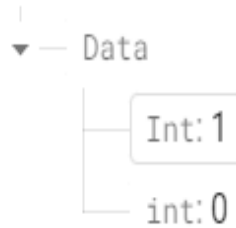
Dijelaskan bahwa gambar diatas MJPG ialah Motion JPEG, yang merupakan format kompresi video yang menggunakan setiap frame video sebagai gambar JPEG terpisah. Dalam konteks Anda, "3774B" mengindikasikan ukuran file dalam byte (B) dari suatu video yang dikompresi menggunakan format MJPG. "22ms" merujuk pada waktu yang diperlukan untuk memproses satu frame video, diukur dalam milidetik (ms). "45.5fps" mengacu pada jumlah frame per detik (fps) yang dapat diproses dalam video tersebut. Lalu "AVG: 19ms (52.6fps)" adalah rata-rata waktu pemrosesan frame dalam gambar diatas.

3.2 Pengujian Software

Pengujian software merupakan tahap penting dalam siklus pengembangan perangkat lunak. Uji software sendiri meliputi pengujian aplikasi yang nantinya digunakan sebagai alat pemantau dirumah. Pengujian Software untuk memastikan apakah keseluruhan bagian dari sistem yang akan kita pergunakan nanti bekerja secara efektif?

1) Pengujian Penampilan Data Pada Firebase

Langkah selanjutnya merupakan tahap perangkat untuk menampilkan pembacaan detektor yang ditangkap oleh sensor PIR pada Database Firebase. Tahap pengujian ini menunjukkan data analog yang sebelumnya dikirim ke Firebase melalui Hardware.

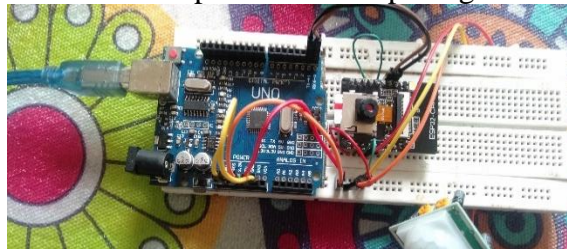


Gambar 14 Hasil pada Firebase

Gambar diatas menunjukkan bahwa Hasil dari data yang didapatkan dari sensor PIR berupa data tersebut hasil data analog yang ditampilkan pada android itu sendiri.

3.3 Pengujian Keseluruhan Sistem

Hal ini merupakan tahap penting untuk memastikan bahwa semua komponen dan fungsi sistem berfungsi dengan baik dan berintegrasi dengan benar. ada beberapa Hardware seperti Arduino Uno, sensor pir HCSR501, dll di mana dapat menguji perangkat keras pada tahap awal hingga akhir ini. Di bawah ini adalah representasi dari perangkat keras yang digunakan ;



Gambar 15 Alat yang digunakan

Gambar yang ditampilkan diatas merupakan alat alat yang digunakan untuk mendeteksi penyusup di rumah. Dengan menyusun perangkat-perangkat tersebut menjadi satu rangkaian sehingga dapat bekerja dan mengirimkan data ke perangkat tersebut, maka keamanan dirumah masih bisa dipantau apabila nantinya ada pergerakan maka rangkaian tersebut akan dapat membaca pergerakan tersebut dan mengeluarkan output berupa data analog. Di bawah ini adalah cuplikan program yang digunakan pada perangkat ini.

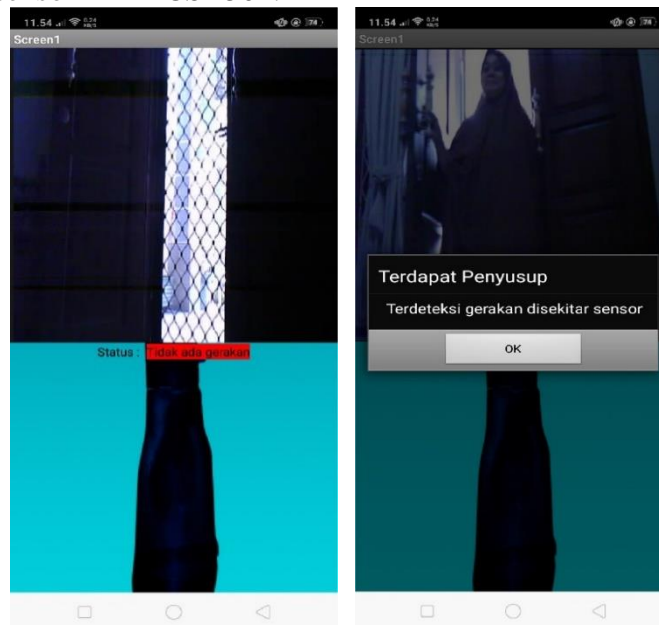
```

isObstacle = digitalRead(isObstaclePin);
if (isObstacle == LOW) {
Serial.println("Depan hanya kosong");
count = 0;
Serial.printf("Data \n", Firebase.setInt(fbdo, "/Data/int", count));
Serial.printf("Data saat ini : %s\n", Firebase.getInt(fbdo, "/Data/int") ? String(fbdo.toInt()).c_str() : fbdo.errorReason().c_str());
delay(1000);
//digitalWrite(LED_BUILTIN, HIGH);
} else {
Serial.println("Terdeteksi di depan!!");
count = 1;
Serial.printf("Data \n", Firebase.setInt(fbdo, "/Data/int", count));
Serial.printf("Data saat ini : %s\n", Firebase.getInt(fbdo, "/Data/int") ? String(fbdo.toInt()).c_str() : fbdo.errorReason().c_str());
delay(1000);
// digitalWrite(LED_BUILTIN, LOW);
}
delay(200);
delay(10000);
}

```

Gambar 16 source code pengujian

Gambar diatas memperlihatkan bagian dari program yang digunakan untuk memproses pembacaan sensor. Selain itu kode ini memiliki beberapa fungsi seperti : B. pengiriman data ke Firebase dan pengiriman data ke aplikasi Android. Setelah data dikirim ke Firebase, langkah selanjutnya adalah menampilkan data pada aplikasi di ponsel. Kode sumber menjelaskan perintah yang mengirim data ke Firebase, sebuah fitur yang membantu Anda melihat hasil yang Anda dapatkan dari sensor PIR HCSR501.



Gambar 17 Tampilan Hasil Ketika tidak ada Gerakan /ada gerakan

Pada gambar di atas memperlihatkan hasil akhir dari aplikasi ini. Fitur ini akan digunakan sebagai antarmuka untuk memonitor keamanan rumah. Untuk mengakses aplikasi, unduh aplikasinya terlebih dahulu.

Setelah dilakukan uji coba baik dari segi perangkat Hardware maupun perangkat software yang dipergunakan, sistem ini dapat berjalan sesuai dengan yang diharapkan. Sehingga sudah dapat dikatakan bahwa Prototype sistem monitoring keamanan menggunakan alarm berbasis IoT ini dapat bekerja dengan baik. Dengan tampilan aplikasi yang sangat sederhana, tentunya akan mempermudah dalam melakukan pemantauan sekitar lingkungan. Pada tahap penelitian, ketika melakukan sebuah penelitian, maka akan dievaluasi dari segi kelebihan dan kekurangannya. Semua jenis manfaat bagi penulis dan pembaca diperhitungkan. Segala kekurangan dan kelebihan dalam penelitian ini diharapkan dapat memunculkan ide yang lebih baik lagi untuk penelitian selanjutnya.

4. KESIMPULAN

Hasil dan perancangan analisa hasil penelitian, bisa disimpulkan beberapa hal antara lain:

- 1) Arduino Uno dapat berperan sebagai pengendali mikrokontroler yang mengontrol dan mengintegrasikan ESP32-CAM dan sensor PIR.
- 2) Sensor PIR yang digunakan merupakan jenis module sensor HCSR501. Sensor ini memiliki sensitivitas yang dapat disesuaikan melalui potensiometer yang ada di modulnya, memungkinkan penyesuaian tingkat sensitivitas pada deteksi gerakan.
- 3) Sensor PIR Sensor ini biasanya bekerja dalam rentang jarak tertentu, sehingga perlu memperhatikan jarak dan sudut pandang yang optimal untuk mendapatkan hasil deteksi yang akurat.

5. DAFTAR PUSTAKA

- Ali, Z. H., Ali, H. A., & Badawy, M. M. (October 2015). Internet of Things (IoT): Definitions, Challenges and Recent Research Directions. *International Journal of Computer Applications* (0975 – 8887), 128.
- Dismawan, M. A. (2019). RANCANG BANGUN SISTEM PENGENDALIAN DAN MONITORING KEAMANAN PINTU RUMAH.
- E, A., W, S., & A, A. (2018). Pemanfaatan Realtime Database di Platform Firebase Pada Aplikasi E-Tourism Kabupaten Nabire. 22(1), 20–26.
- Gifson, A. (n.d.). SISTEM PEMANTAU RUANG JARAK JAUH DENGAN SENSOR PASSIVE INFRARED BERBASIS MIKROKONTROLER AT89S52. ISSN: 1693-6930.
- Iyapo, K. O., Fasanla, O. M., Egbuwalo, S. A., Akinbobola, A. J., & Oni, O. T. (2018). DESIGN AND IMPLEMENTATION OF MOTION DETECTION ALARM AND. *International Journal of Engineering and Advanced Technology Studies*.
- Juditha, C. (2019). Buzzer di Media Sosial Pada Pilkada dan Pemilu Indonesia Buzzer in Social Media in Local Elections and Indonesian Elections. *Prosiding Seminar Nasional Komunikasi dan Informatika #3*, 199-212.
- Maulana, I. F. (2021). Penerapan Firebase Realtime Database pada Aplikasi E-Tilang Smartphone berbasis Mobile Android. *Rekayasa Sistem dan Teknologi Informasi*.
- Ramschie, A., Maka, J., Katuuk, R., & Ponggawa, V. (2021). Pemanfaatan ESP32 Pada Sistem

Keamanan Rumah Tinggal Berbasis. Prosiding The 12th Industrial Research Workshop and National Seminar.