

RANCANG & BANGUN BEL RUMAH OTOMATIS BERBASIS IOT DAN FACE RECOGNITION DENGAN APLIKASI ANDROID MENGGUNAKAN ESP32-CAM

Rizki Amelia Lestari¹

¹Teknik Informatika, STMIK “AMIKBANDUNG”

e-mail: ¹rizkiamelia843@gmail.com

ABSTRAK

Sebagian masyarakat biasa melakukan kegiatan bertamu untuk mempererat tali silaturahmi dengan orang lain. Namun saat tamu memasuki area rumah dan tiba di pintu masuk, terkadang penghuni rumah sedang tidak ada di rumah atau tidak mengetahui jika ada tamu yang sedang berkunjung. Selain itu jika penghuni rumah sedang beraktifitas di dalam rumah dan berada jauh dari pintu masuk rumah membuat suara tamu tidak terdengar. Untuk mengatasi permasalahan di atas maka di zaman sekarang ini dapat dimanfaatkan teknologi dalam bidang *internet of things* (IoT) yaitu untuk pembuatan *prototype* bel rumah otomatis berbasis IoT dan *face recognition* dengan menggunakan modul ESP32-CAM serta aplikasi Android sebagai media notifikasi. *Prototype* bel rumah otomatis ini dibangun menggunakan komponen ESP32-CAM, sensor PIR, buzzer serta API *face recognition* sebagai *database* pengenalan wajah. Untuk pembangunan *prototype* aplikasi Android sebagai media notifikasi menggunakan software Android Studio dan Firebase sebagai basis datanya. Adapun metode yang digunakan yaitu metode *prototyping*. Berdasarkan hasil pengujian yang telah dilakukan, diketahui bahwa sensor PIR mampu mendeteksi tamu dengan jarak kurang lebih 200 meter dan ESP32-CAM dapat melakukan deteksi wajah dengan rentang jarak 40-80 cm sedangkan persentase error fungsional sebesar 16.6%. Hal ini menunjukkan bahwa hasil pengujian pada bel rumah otomatis bisa berjalan baik.

Kata Kunci: IoT, *face recognition*, bel rumah otomatis, ESP32-CAM, aplikasi Android

1. PENDAHULUAN

Menurut Badan Pusat Statistik indeks kebahagiaan Indonesia tahun 2021 didominasi oleh aspek-aspek diantaranya: 1) keharmonisan keluarga (82.56%), 2) keadaan lingkungan (81.56%), 3) kondisi keamanan (81.20%), 4) hubungan sosial (79.10%) dan 5) kesehatan (76.29%) [1]. Dari data tersebut dapat dilihat jika hubungan sosial merupakan aspek yang berpengaruh dalam mempengaruhi tingkat kebahagiaan di Indonesia. Salah satu cara dalam melaksanakan hubungan sosial adalah dengan bertamu. Bertamu merupakan kegiatan berkunjung ke rumah orang lain dalam rangka mempererat tali silaturahmi [2]. Dalam kegiatan bertamu biasanya tamu akan mengetuk pintu, menekan bel atau mengucapkan salam untuk izin masuk rumah orang lain.

Namun saat tamu berkunjung terkadang penghuni rumah sedang tidak ada di rumah dan penghuni rumah tidak tahu jika ada tamu yang sempat berkunjung ke rumah. Selain itu suara tamu yang kurang keras saat mengetuk pintu atau mengucapkan salam mengakibatkan penghuni rumah sedikit lebih lama untuk membukakan pintu. Sehingga diperlukan bel rumah otomatis dengan fitur notifikasi ke penghuni rumah baik saat di rumah maupun di luar rumah.

Telah dilakukan penelitian terkait Bel Rumah Otomatis berbasis IoT menggunakan sensor PIR dan NodeMCU. Sujono dan Aqilah (2021) [3] dalam penelitiannya membuat bel rumah otomatis berbasis IoT menggunakan modul mini sensor PIR, NodeMCU, Buzzer dan aplikasi Blynk untuk notifikasi pengguna. Sensor PIR akan mendeteksi pengunjung lalu data akan diolah oleh NodeMCU, yang nantinya akan mengeksekusi Blynk untuk mengirimkan notifikasi dan buzzer untuk mengeluarkan suara. Hasilnya bel rumah akan berbunyi jika jarak pengunjung dengan pintu > 30 cm dan jika lebar sudut sensor dengan pengunjung rata-rata sudut 60°.

Selanjutnya Plasida (2021) [4] dalam penelitiannya mengenai sistem pemantau kedatangan tamu berbasis internet of things (IoT) menggunakan modul ESP32-CAM, infrared sensor (IR), buzzer, solenoid doorlock, FTDI, relay dan chat bot aplikasi telegram untuk pengendali pengguna. Sensor IR akan mendeteksi telapak tangan pengguna yang mendekat dengan jarak 0,5 – 3 cm sebagai tombol touchless kemudian akan dilakukan foto otomatis yang nantinya dikirimkan ke chat bot aplikasi Telegram. Pengguna dapat melakukan pengendalian pada pintu seperti buka kunci pintu, tutup kunci pintu dan foto otomatis lewat chat bot aplikasi Telegram. Hasilnya informasi berupa gambar dan

chatting yang dikirimkan pengguna melalui chat bot Telegram dapat diterima oleh ESP32-CAM dalam waktu kurang dari 3 detik sehingga dapat digunakan untuk memantau kedatangan tamu secara real-time.

Berdasarkan penelitian yang dilakukan, maka penelitian ini bertujuan untuk mengembangkan bel rumah otomatis berbasis IoT menggunakan ESP32-CAM dengan aplikasi Android sebagai media monitoring dan notifikasi.

3. METODE PENELITIAN

Penelitian ini menggunakan metode prototyping dimana terdapat 5 tahapan diantaranya 1) Communication, 2) Quick Plan, 3) Modelling Quick Design, 4) Construction, 5) Deployment Delivery & Feedback. Menurut Noor Santi (2018) prototyping merupakan suatu metode pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai [5]. Berikut merupakan penerapan metode prototyping pada penelitian ini :

3.1 Communication

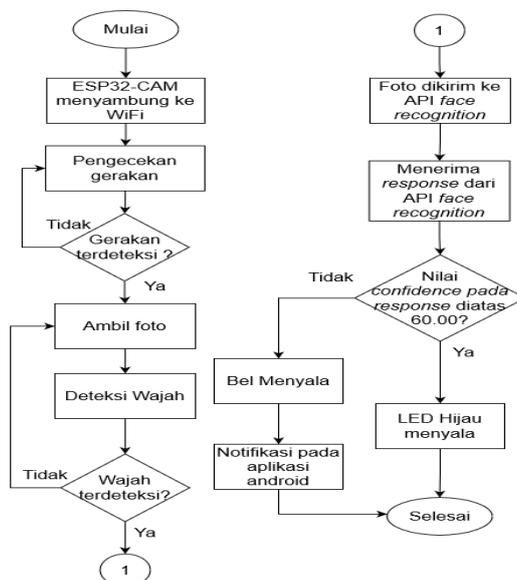
Pada tahap ini dilakukan proses analisis dari data yang didapatkan. Teknik pengumpulan data berupa observasi (pengamatan) dan studi literatur (mempelajari referensi). Dari teknik pengumpulan data yang dilakukan didapatkan permasalahan dimana saat tamu berkunjung, penghuni rumah sedang tidak ada dirumah sehingga penghuni rumah tidak tahu siapa saja tamu yang berkunjung saat penghuni rumah tidak dirumah. Serta penghuni rumah yang terkadang tidak mendengar ada suara tamu yang berkunjung dikarenakan sedang berada di ruangan yang jauh dari pintu masuk atau sedang beraktifitas di dalam rumah.

3.2 Quick Plan

Pada tahap ini dilakukan perencanaan terhadap identifikasi masalah yang ditemukan sebelumnya. Bel rumah yang dikembangkan berbasis IoT dan *face recognition* dengan modul utama ESP32-CAM. *Flowchart* dimulai dengan menghubungkan modul ESP32-CAM dengan WiFi yang tersedia, kemudian sensor PIR akan mendeteksi gerakan.

Jika terdeteksi maka akan dilakukan ambil foto otomatis dan pengecekan apakah terdeteksi wajah melalui foto. Jika terdeteksi wajah maka foto akan dikirim ke *API face recognition* untuk pengecekan apakah wajah dikenali sebagai penghuni rumah. Jika wajah tidak terdeteksi maka akan dilakukan ambil foto ulang.

Kemudian *API face recognition* mengembalikan *response* dengan nilai *confidence*, apabila nilai *confidence* dibawah 60.00 maka wajah dikenali sebagai tamu yang *output* nya berupa bel yang menyala dan notifikasi pada aplikasi Android. Apabila nilai *confidence* diatas 60.00 maka wajah dikenali sebagai penghuni rumah yang *output* nya LED hijau menyala. Alur kerja yang direncanakan ditunjukkan pada gambar 1.



Gambar 1 Flowchart Bel Rumah Otomatis

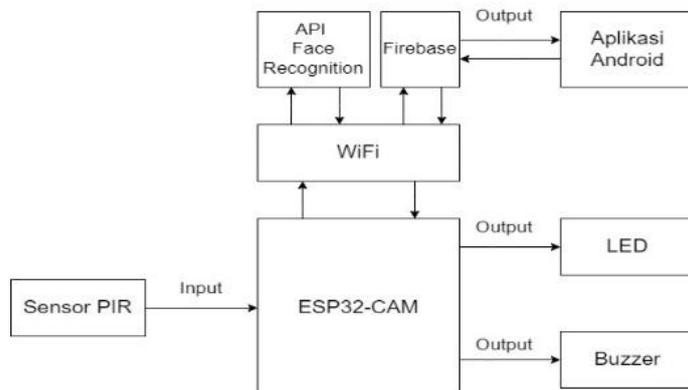
3.3 Modelling Quick Design

Pada tahap ini dilakukan perancangan desain terhadap prototype *hardware* dan *software* bel otomatis berbasis IoT dan *face recognition*. Berikut merupakan perancangan yang dilakukan :

3.3.1 Perancangan Hardware

a. Diagram Blok

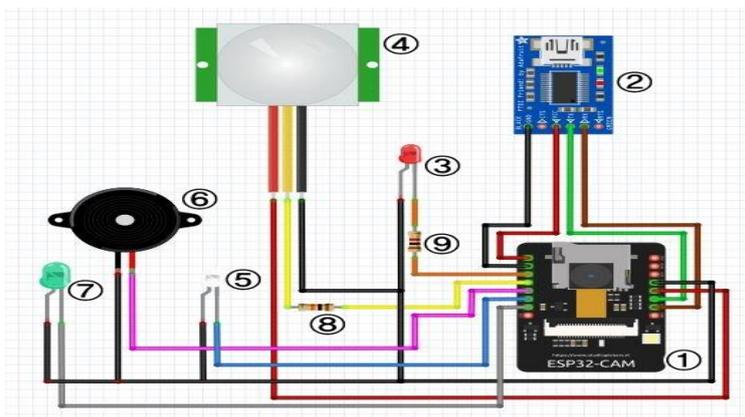
Diagram blok ditunjukkan pada gambar 2 berikut.



Gambar 2 Diagram Blok

b. Rangkaian Hardware

Rangkaian hardware bel rumah otomatis berbasis IoT dan *face recognition* ditunjukkan pada gambar 3 berikut.



Gambar 3 Rangkaian Keseluruhan

Perancangan rangkaian hardware keseluruhan bel rumah otomatis berbasis IoT dan *face recognition* meliputi komponen modul 1) ESP32-CAM, 2) FTDI, 3) LED 5mm Merah, 4) Sensor PIR, 5) LED 5mm Putih, 6) Buzzer, 7) LED 5mm Hijau, 8) Resistor 10K ohm dan 9) Resistor 1K ohm. Daftar koneksi masing-masing modul terhadap pin modul ESP32-CAM ditunjukkan pada tabel 1.

Tabel 1 Daftar Koneksi Pin ESP32-CAM

| | PIN ESP32-CAM | | | | | | | | | |
|------------|---------------|-----|------|-----|----|----|---|--------|-------|-------|
| | 5V | GND | GPIO | | | | | 3.3/5V | U0TXD | U0RXD |
| | | | 12 | 13 | 15 | 14 | 2 | | | |
| FTDI | VCC | GND | | | | | | | RX | TX |
| Sensor PIR | | - | | OUT | | | | + | | |
| Buzzer | | - | | | + | | | | | |
| LED Merah | | - | + | | | | | | | |
| LED Putih | | - | | | | + | | | | |
| LED Hijau | | - | | | | | + | | | |

3.3.2 Perancangan Software

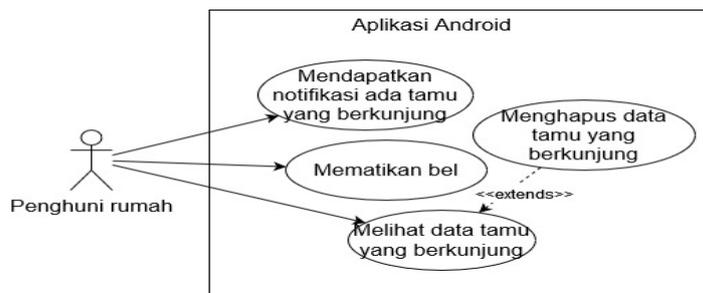
a. Perancangan Aplikasi Android

Fitur yang akan dibangun pada aplikasi Android ditunjukkan pada tabel 2 berikut.

Tabel 2 Kebutuhan Fungsional Aplikasi Android

| Nomor KF | Deskripsi |
|----------|---|
| Aktor 1 | Penghuni Rumah |
| KF-001 | Melihat data tamu |
| KF-002 | Menghapus data tamu |
| KF-003 | Mematikan bel |
| KF-004 | Mendapatkan Notifikasi ada tamu yang berkunjung |

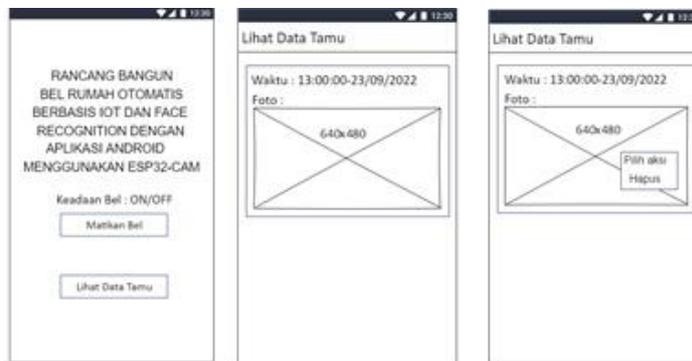
Dari tabel kebutuhan fungsional aplikasi tersebut diperjelas dengan menggunakan *use case diagram* seperti yang ditunjukkan pada gambar 4 berikut.



Gambar 4 Use Case Diagram Kebutuhan Fungsional Aplikasi

Pada gambar diatas terdapat 1 aktor yaitu penghuni rumah yang merupakan aktor utama yang dapat melakukan seluruh aktivitas / *case* dalam aplikasi ini. Aktor penghuni rumah merupakan penghuni rumah yang menggunakan aplikasi Android ini untuk memonitoring siapa saja tamu yang berkunjung. Terdapat 4 aktivitas / *case* yang dapat dilakukan oleh aktor diantaranya mendapatkan notifikasi ada tamu yang berkunjung, mematikan bel, melihat data tamu yang berkunjung dan menghapus data tamu yang berkunjung.

Kemudian selanjutnya dilakukan perancangan tampilan untuk aplikasi Android yang akan dibangun. Perancangan tampilan aplikasi Android ditunjukkan pada gambar 5 berikut.



Gambar 5 Perancangan Tampilan Aplikasi Android

b. Perancangan dengan Arduino IDE

Arduino IDE digunakan sebagai media untuk melakukan proses koding / pemrograman ke modul ESP32-CAM. Pada tahap ini dilakukan perancangan pemrograman yang akan diupload ke modul ESP32-CAM. Perancangan pemrograman disesuaikan dengan alur perencanaan pada tahap *quick plan* sebelumnya.

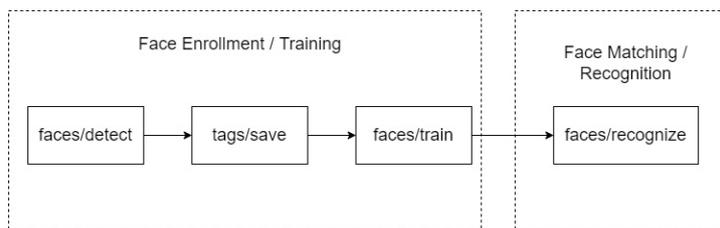
3.4 Construction

Pada tahap ini dilakukan pembangunan bel rumah otomatis baik dari segi software maupun hardware nya. Berikut merupakan tahap pembangunan bel rumah otomatis yang dilakukan :

3.4.1 Konfigurasi API Face Recognition

API *face recognition* dilakukan konfigurasi dengan 2 tahapan utama. *Face enrollment* atau *Training* merupakan tahap untuk mengupload foto wajah yang didaftarkan ke *database* yang kemudian dilakukan proses training melalui API *face recognition*. *Face matching* atau *Recognition* merupakan

tahap untuk melakukan proses *face detection* yang diterapkan ke ESP32-CAM. Alur konfigurasi API *face recognition* ditunjukkan pada gambar 6 berikut.



Gambar 6 Tahapan Konfigurasi API Face Recognition

3.4.2 Pemasangan Rangkaian Komponen

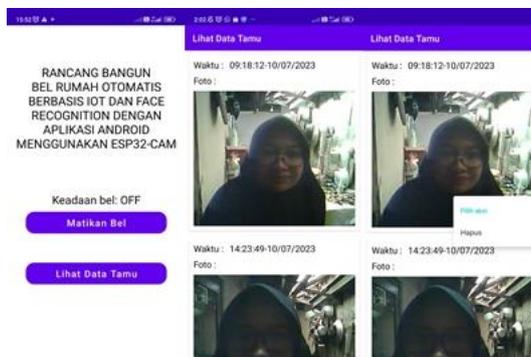
Pemasangan rangkaian komponen ke dalam wadah box bel otomatis. Pemasangan dilakukan sesuai dengan perancangan rancangan hardware yang sudah dilakukan sebelumnya. Pemasangan rangkaian komponen ditunjukkan pada gambar 7 berikut.



Gambar 7 Tahapan Pemasangan Rangkaian Komponen

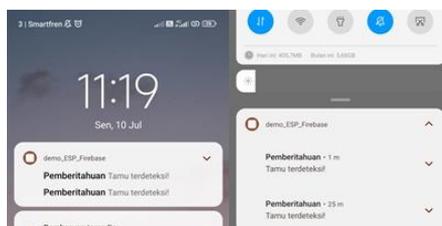
3.4.3 Tampilan Aplikasi Android

Aplikasi Android dibangun sesuai dengan perancangan tampilan aplikasi Android yang sudah dibuat sebelumnya. Tampilan aplikasi Android ditunjukkan pada gambar 8 berikut.



Gambar 8 Tampilan Aplikasi Android

Serta tampilan notifikasi aplikasi Android yang ditunjukkan pada gambar 9 berikut.



Gambar 9 Tampilan Notifikasi Aplikasi Android

3.5 Deployment, Delivery & Feedback

Pada tahap ini dilakukan pemasangan alat yang sudah dirangkai pada pintu rumah. Kemudian dilakukan pengujian singkat terhadap alat sebelum nantinya dilakukan pengujian keseluruhan. Pemasangan alat pada pintu rumah ditunjukkan pada gambar 10 berikut.



Gambar 10 Tahapan Pemasangan Alat Pada Pintu

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Sensor PIR berdasarkan Jarak

Pada pengujian ini dilakukan uji coba sensor PIR terhadap objek dengan jarak tertentu. Sensor PIR sudah diatur sensitivitas nya dengan memutar potensiometer sebelah kanan / jarak ke arah kiri penuh dan potensiometer sebelah kiri / waktu ke arah kiri penuh. Objek yang diuji memiliki tinggi sekitar 155-160 cm. Pengujian sensor PIR berdasarkan jarak ditunjukkan pada tabel 3 berikut.

Tabel 3 Pengujian Sensor PIR Berdasarkan Jarak

| Objek | Tinggi Objek | Jarak | | |
|------------|--------------|--------|--------|--------|
| | | 300 cm | 200 cm | 100 cm |
| Penghuni 1 | 158 cm | x | √ | √ |
| Penghuni 2 | 155 cm | x | √ | √ |
| Penghuni 3 | 157 cm | x | √ | √ |
| Tamu 1 | 159 cm | x | √ | √ |
| Tamu 2 | 155 cm | x | √ | √ |
| Tamu 3 | 160 cm | x | √ | √ |

Keterangan:

√ = sensor PIR aktif

x = sensor PIR tidak aktif

Dari hasil tersebut didapatkan kesimpulan bahwa sensor PIR akan aktif jika objek pada rentang jarak dibawah kurang lebih 200 cm.

4.2 Pengujian Deteksi Wajah Berdasarkan Jarak

Pada pengujian ini dilakukan uji coba kamera terhadap wajah objek dengan jarak tertentu. Deteksi wajah ini dapat dilakukan jika sensor PIR mendeteksi adanya gerakan. Objek yang disertakan yaitu penghuni dan tamu dengan rentang tinggi badan 155-160 cm. Pengujian deteksi wajah berdasarkan jarak ditunjukkan pada tabel 4 berikut.

Tabel 4 Pengujian Deteksi Wajah Berdasarkan Jarak

| Objek | Tinggi Objek | Jarak Wajah Ke Kamera | | | | |
|------------|--------------|-----------------------|-------|-------|-------|-------|
| | | 100 cm | 80 cm | 60 cm | 40 cm | 20 cm |
| Penghuni 1 | 158 cm | x | √ | √ | √ | x |
| Penghuni 2 | 155 cm | x | x | √ | √ | x |
| Penghuni 3 | 157 cm | x | √ | √ | √ | x |
| Tamu 1 | 159 cm | x | √ | √ | √ | x |
| Tamu 2 | 155 cm | x | x | √ | √ | x |
| Tamu 3 | 160 cm | x | √ | √ | √ | x |

Keterangan :

√ = *face detection* berhasil dijalankan

x = *face detection* gagal dijalankan

Dari hasil tersebut didapatkan bahwa *face detection* berhasil dijalankan jika jarak wajah dengan kamera pada rentang 40 – 80 cm.

4.3 Pengujian Deteksi Wajah Berdasarkan Kondisi Cahaya

Pada pengujian ini dilakukan uji coba deteksi wajah dengan kondisi cahaya terang dan gelap. Kondisi terang artinya kondisi area tempat bel otomatis dipasang dalam kondisi pencahayaan yang cukup dengan lampu, kondisi gelap artinya kondisi area tempat bel otomatis dipasang dalam kondisi pencahayaan yang kurang tanpa lampu. Pengujian deteksi wajah berdasarkan kondisi cahaya ditunjukkan pada tabel 5 berikut.

Tabel 5 Pengujian Deteksi Wajah Berdasarkan Kondisi Cahaya

| Objek | Kondisi Cahaya | |
|------------|----------------|-------|
| | Terang | Gelap |
| Penghuni 1 | √ | X |
| Penghuni 2 | √ | X |
| Penghuni 3 | √ | X |
| Tamu 1 | √ | X |
| Tamu 2 | √ | X |
| Tamu 3 | √ | X |

Keterangan :

√ = *face detection* berhasil dijalankan

x = *face detection* gagal dijalankan

Dari hasil tersebut didapatkan bahwa *face detection* berhasil dijalankan jika cahaya dalam kondisi terang dan gagal jika cahaya dalam kondisi gelap.

4.4 Pengujian Pengenalan Wajah

Pada pengujian ini dilakukan uji coba pengenalan wajah. Pengenalan wajah dilakukan apabila deteksi wajah berhasil dijalankan. Pengenalan wajah dibagi menjadi 2 kategori yaitu penghuni rumah dan tamu. Penghuni rumah merupakan data wajah yang terdaftar di *database API face recognition* dan tamu merupakan data wajah yang tidak terdaftar di *database API face recognition*. Pengujian pengenalan wajah ditunjukkan pada tabel 6 berikut.

Tabel 6 Pengujian Pengenalan Wajah

| Objek | Wajah dikenali sebagai penghuni rumah |
|------------|---------------------------------------|
| Penghuni 1 | √ |
| Penghuni 2 | √ |
| Penghuni 3 | √ |
| Tamu 1 | √ |
| Tamu 2 | X |
| Tamu 3 | X |

Keterangan :

√ = wajah dikenali sebagai penghuni rumah

x = wajah dikenali sebagai tamu

Pada objek tamu 1 wajah dikenali sebagai penghuni rumah, hal ini karena wajah objek memiliki kemiripan dengan wajah objek penghuni rumah sehingga dalam *response* pada *API face recognition* persentase *confidence* yang didapat yaitu 64.00 sehingga tamu 1 diindikasikan dengan penghuni rumah. Dari hasil tersebut didapatkan bahwa pengenalan wajah bisa dijalankan walaupun terdapat kelemahan yaitu pada wajah tamu yang mirip dengan penghuni rumah sehingga dideteksi sebagai penghuni rumah oleh pengenalan wajah ini.

4.5 Pengujian Fungsionalitas Aplikasi

Pada tahap ini dilakukan pengujian pada aplikasi Android. Pengujian fungsionalitas aplikasi ditunjukkan pada tabel 7 berikut.

Tabel 7 Pengujian Fungsionalitas Aplikasi

| Fungsi | Hasil yang diharapkan | Hasil pengujian | Keterangan |
|-------------------|-------------------------------------|---|------------|
| Melihat data tamu | Dapat mengambil data dari data tamu | Berhasil mendapatkan data tamu dan menampilkan data tamu pada halaman lihat data tamu | Sesuai |

| | | | |
|---------------------|--|--|--------|
| Menghapus data tamu | Dapat menghapus data dari data tamu | Berhasil menghapus data tamu dan menampilkan toast data dihapus pada halaman lihat data tamu | Sesuai |
| Mematikan bel | Dapat mematikan bel pada alat | Berhasil memperbarui data kondisi bel menjadi OFF | Sesuai |
| Melihat notifikasi | Dapat mendapatkan notifikasi yang di trigger oleh alat | Berhasil menampilkan notifikasi dari <i>trigger</i> alat | Sesuai |

4.6 Pengujian Seluruh Fungsionalitas

Pengujian ini dilakukan untuk memastikan apakah alat yang sudah dibuat sesuai dengan perencanaan yang dibuat sebelumnya, yaitu bel rumah otomatis berbasis IoT dengan *face recognition* yang keluarannya jika terdeteksi sebagai tamu maka bel rumah akan menyala dan notifikasi akan dikirim ke aplikasi Android. Pengujian ditunjukkan pada tabel 8 berikut.

Tabel 8 Pengujian Seluruh Fungsionalitas

| Objek | Jumlah Pengujian | | Total (Sesuai) |
|------------|------------------|----|----------------|
| | 1x | 2x | |
| Penghuni 1 | √ | √ | 2/2 |
| Penghuni 2 | √ | √ | 2/2 |
| Penghuni 3 | √ | √ | 2/2 |
| Tamu 1 | X | x | 0/2 |
| Tamu 2 | √ | √ | 2/2 |
| Tamu 3 | √ | √ | 2/2 |

Keterangan :

√ : Sesuai harapan

x : Tidak sesuai harapan

Pada objek tamu 1 dilakukan 2 kali pengujian dan hasilnya 2 pengujian tersebut tidak sesuai harapan, hal ini dikarenakan objek tamu 1 terdeteksi sebagai penghuni rumah sehingga tidak sesuai harapan. Selanjutnya akan dilakukan perhitungan error menggunakan rumus persentase error [6]. Rumus persentase error ditunjukkan pada rumus (1)

$$Persen\ error = \frac{[nilai\ yang\ diterima - nilai\ eksperimental]}{nilai\ yang\ diterima} \times 100\% \tag{1}$$

Keterangan :

Nilai eksperimental = jumlah nilai berhasil yang didapatkan saat pengujian

Nilai yang diterima = jumlah pengujian yang dilakukan saat pengujian

Terdapat 2 kali pengujian dengan 6 objek maka pengujian keseluruhan dilakukan sebanyak 12 kali. Dari 12 kali pengujian tersebut didapatkan hasil pengujian berhasil sebanyak 10 pengujian dan hasil pengujian tidak berhasil sebanyak 2 pengujian. Perhitungan persentase error ditunjukkan pada rumus 1 berikut.

$$Persen\ error = \frac{12-10}{12} \times 100\% = 16.6\% \tag{1}$$

Dari hasil perhitungan menggunakan 6 wajah dengan masing-masing pengujian sebanyak 2 kali menghasilkan nilai error sebanyak 16.6%. didapatkan kelemahan dari alat ini yaitu pada wajah tamu yang mirip dengan penghuni rumah sehingga dideteksi sebagai penghuni rumah padahal itu merupakan tamu serta tidak bisa membedakan wajah asli dan wajah di foto.

5. KESIMPULAN

Pada pembuatan bel otomatis berbasis IoT dan *face recognition* dengan aplikasi Android menggunakan ESP32-CAM didapatkan kesimpulan sebagai berikut. Berdasarkan hasil uji coba pada alat yang dibangun maka penghuni rumah dapat mengetahui ada tamu yang berkunjung meskipun pada saat penghuni rumah sedang ada di rumah atau di luar rumah karena bel rumah otomatis tersebut mempunyai fitur notifikasi lewat bel rumah dan aplikasi Android. Hal ini membuktikan bahwa seluruh

fungsionalitas menghasilkan persentase error sebesar 16.6 %. Dengan demikian apabila tamu terdeteksi maka bel rumah akan menyala dan notifikasi akan diterima oleh penghuni rumah melalui aplikasi Android. Penghuni rumah dapat melakukan monitoring tamu yang berkunjung saat penghuni rumah sedang ada di rumah atau di luar rumah karena bel rumah otomatis yang dibangun mempunyai fitur untuk melihat data tamu yang berkunjung melalui aplikasi Android. Hal ini terbukti dari hasil pengujian fungsionalitas pada aplikasi, pemilik rumah dapat melakukan pengolahan data seperti melihat, menghapus, mendapatkan notifikasi, mematikan bel. Dengan demikian penghuni rumah dapat melakukan monitoring tamu yang berkunjung secara mudah.

DAFTAR PUSTAKA

- [1] B. P. Statistik, "Dimensi Kepuasan Hidup Indeks Kebahagiaan 2021," Badan Pusat Statistik, 11 Januari 2022. [Online]. Available: <https://www.bps.go.id/indicator/34/627/1/dimensi-kepuasan-hidup-indeks-kebahagiaan.html>. [Accessed 19 Juni 2023].
- [2] D. Charilisyah, "Mengajarkan Tata Cara Bertamu Kepada Anak Usia Dini (Untuk Guru dan Orang Tua)," *EDUCHILD*, vol. 5, no. 2, p. 154, 2016.
- [3] S. and E. Aqilah, "Bel Rumah Otomatis Berbasis IoT menggunakan Sensor PIR dan NodeMCU," *Exact Paper in Compilation (EPIC)*, vol. 3, no. 3, pp. 383-390, 2021.
- [4] P. A. A. P. Basabilik, "Rancang Bangun Sistem Pemantau Kedatangan Tamu Berbasis Internet Of Things (Iot)," *Prina Fisika*, vol. 9, no. 2, pp. 110-116, 2021.
- [5] M. Prabowo, *Metodologi Pengembangan Sistem Informasi*, Salatiga: Lembaga Penelitian dan Pengabdian kepada Masyarakat (LP2M) IAIN Salatiga, 2020.
- [6] A. M. Helmenstine, "How to Calculate Percent Error," ThoughtCo., 2 November 2020. [Online]. Available: https://www-thoughtco-com.translate.goog/how-to-calculate-percent-error-609584?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc. [Accessed 15 Juli 2023].
- [7] M. F. Wicaksono and M. D. Rahmafya, "Implementasi Arduino dan ESP32-CAM untuk Smart Home," *JATI*, vol. 10, no. 1, pp. 41-50, 2020.
- [8] Y. Yudhanto and A. Aziz, *Pengantar Teknologi Internet of Things*, Surakarta: UNS Press, 2019.
- [9] A. Suryansyah, R. Habibi and R. M. Awangga, *Penggunaan Face Recognition untuk Akses Ruang*, Bandung: Kreatif Industri Nusantara, 2020.
- [10] D. "DFR0602_Web.pdf," 15 August 2019. [Online]. Available: https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/DFR0602_Web.pdf. [Accessed 30 May 2023].
- [11] N. Sfaat, *ANDROID Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*, Bandung: Penerbit INFORMATIKA, 2015.