



## SMART DOOR LOCK SYSTEM BERBASIS APLIKASI BLYNK DENGAN FACE RECOGNITION MENGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)

Retno Ayuni<sup>1</sup>, Phitsa Mauliana<sup>2</sup>

<sup>1,2</sup>Universitas Adhirajasa Reswara Sanjaya

Jl. Terusan Sekolah No.1-2, Cicaheum, Kec. Kiaracondong, Kota Bandung, Indonesia

<sup>1</sup>[rayuni20021020@gmail.com](mailto:rayuni20021020@gmail.com)

### Abstract

*Access control to buildings is a crucial aspect in ensuring safety and preventing criminal activities such as theft. The use of conventional physical locks is often considered inadequate as they are prone to tampering and theft. To address this issue, a smart door lock system integrated with the Blynk application and face recognition technology provides an effective solution. This system allows remote door control via a smartphone, replacing physical keys that can be easily compromised. The Blynk application, as an Internet of Things (IoT) platform, offers an interface for control over a Wi-Fi network, while face recognition adds an extra layer of security by verifying user identity. The Convolutional Neural Network (CNN) method is utilized for face recognition, ensuring high accuracy and providing a higher level of security compared to conventional methods. This study explores the application of IoT-based smart door locks and face recognition in enhancing building security in an efficient and practical manner.*

**Keywords:** *Blynk, Convolutional Neural Network (CNN), Face Recognition, Internet of Things (IoT), Smart Door Lock.*

### Abstrak

Keamanan akses pintu pada bangunan merupakan salah satu aspek penting dalam menjaga keselamatan dan mencegah tindakan kriminal seperti pencurian. Penggunaan kunci fisik yang konvensional sering kali dianggap kurang aman karena rentan terhadap pembobolan. Untuk mengatasi hal ini, sistem kunci pintu pintar (*smart door lock*) yang terintegrasi dengan aplikasi *Blynk* dan teknologi pengenalan wajah (*face recognition*) menjadi solusi yang efektif. Sistem ini memungkinkan pengendalian akses pintu secara jarak jauh menggunakan *smartphone*, menggantikan kunci fisik yang mudah disalahgunakan. Aplikasi *Blynk*, sebagai platform *Internet of Things* (IoT), menyediakan antarmuka kendali melalui jaringan *Wi-Fi*, sementara pengenalan wajah menambah lapisan keamanan dengan memverifikasi identitas pengguna. Metode *Convolutional Neural Network* (CNN) digunakan untuk pengenalan wajah dengan

### Article History:

Received: Maret 2025

Reviewed: Maret 2025

Published: Maret 2025

Plagiarism Checker No 234

Prefix DOI :

10.8734/Kohesi.v1i2.365

Copyright : Author

Publish by : Kohesi



This work is licensed

under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



akurasi tinggi, memberikan tingkat keamanan yang lebih baik dibandingkan dengan metode konvensional. Penelitian ini mengkaji penerapan sistem *smart door lock* berbasis *IoT* dan *face recognition* dalam meningkatkan keamanan bangunan secara efisien dan praktis.

**Kata kunci:** *Blynk*, *Convolutional Neural Network* (CNN), *Internet of Things* (IoT), Pengenalan Wajah, *Smart Door Lock*.

## 1. PENDAHULUAN

Keamanan dan kenyamanan merupakan faktor penting yang perlu diperhatikan ketika seseorang berada di dalam maupun di luar sebuah bangunan. Akses ke sebuah bangunan tidak boleh sembarangan untuk menghindari gangguan [1]. Oleh karena itu, pintu sebagai akses utama keluar masuk perlu diamankan untuk menghindari gangguan dari orang yang tidak diinginkan seperti pencuri [2]. Saat ini proses membuka dan mengunci pintu banyak dilakukan dengan cara manual atau konvensional. Namun, metode ini kurang efektif karena orang yang membuka akses harus membawa kunci. Selain itu, keamanan juga masih belum terjamin karena kunci masih bisa dibobol [3]. Salah satu strategi yang dapat dilakukan adalah dengan mengadopsi teknologi kunci pintar (*smart door lock*).

*Smart door lock* memungkinkan pengendalian pintu dari jarak jauh sehingga memudahkan pemilik bangunan mengatur akses pintu [4]. Teknologi ini juga muncul sebagai alternatif membawa kunci fisik sehingga orang tidak perlu membawa kunci sama sekali untuk membuka pintu. *Smart door lock* juga memiliki berbagai mode akses dan dilengkapi fitur alarm untuk meningkatkan keamanan [5]. *Smart door lock* juga dapat dikolaborasikan dengan *Internet of Things* (IoT) yang dapat meningkatkan keamanan bangunan.

*Internet of Things* (IoT) merupakan konsep untuk memperluas kegunaan konektivitas internet guna berbagi data, mengendalikan perangkat dari jarak jauh, dan fungsi lainnya. Secara prinsip, *IoT* adalah pengembangan dari teknologi jaringan komputer dan internet, yang menghubungkan perangkat seperti sensor dan mikrokontroler ke jaringan *WiFi* untuk menjalankan berbagai fungsi dengan sistem tertanam (*embedded system*) [2]. Penerapan *IoT* dalam keamanan akses pintu sangat relevan karena memungkinkan pengguna untuk mengunci atau membuka pintu secara otomatis dari jarak jauh. Dengan menggunakan *smartphone*, pengguna bisa memastikan keamanan rumah mereka kapan saja dan di mana saja, tanpa perlu berada di lokasi [6].

Pengembangan *IoT* dalam keamanan akses pintu dapat dibantu dengan *Blynk*. Terdapat tiga komponen utama dalam *Blynk*, yaitu: Aplikasi *Blynk* (*Blynk apps*), server *Blynk* (*Blynk server*), dan pustaka *Blynk* (*Blynk libraries*) [7]. *Blynk* dipilih karena dirancang untuk aplikasi pada berbagai sistem operasi OS (iOS dan Android) serta memungkinkan kendali modul seperti *Arduino*, *Raspberry Pi*, ESP8266, WEMOS D1, dan modul sejenisnya melalui internet. *Blynk* mempermudah pembuatan dan pengelolaan antarmuka kendali dari perangkat *smartphone* untuk *smart door lock* serta mampu meningkatkan keamanan dan kenyamanan pengguna [8].



Akses membuka dan mengunci pintu masih memiliki celah disalahgunakan orang lain selain pengguna. Oleh karena itu, fitur *face recognition* ditambahkan sehingga hanya pengguna yang terverifikasi yang dapat mengendalikan akses. Pengenalan wajah (*face recognition*) adalah teknik identifikasi yang mirip dengan pengenalan sidik jari dan retina mata, di mana gambar yang ditangkap oleh kamera akan dibandingkan dengan tekstur dan kontur wajah yang telah terdaftar sebelumnya (Pribadi, 2023). Oleh karena itu, inovasi dalam teknik ini sangat penting untuk meningkatkan keamanan akses pintu dengan memanfaatkan teknologi untuk memudahkan proses identifikasi dan akses [9].

Integrasi *smart door lock*, *Blynk*, dan *face recognition* memberikan manfaat signifikan dalam meningkatkan keamanan dan kenyamanan akses pintu. Sejauh ini beberapa penelitian sudah mengkaji pemanfaatan *IoT* dalam *smart door lock*. Penelitian oleh Sanjaya & Jaya (2023) menunjukkan *Blynk* dapat digunakan untuk menampilkan hasil pembacaan sensor dan mengontrol fungsi *on/off* lampu, pintu, dan kipas angin berdasarkan pembacaan sensor. Penelitian lainnya oleh Fadly dkk. (2021) memanfaatkan *face recognition* untuk menjaga keamanan pintu dan mencegah pencurian di kamar kos [9]. Meski demikian, waktu respons masih terbilang lama yaitu sekitar 6 hingga 8 detik.

Penelitian ini mengintegrasikan *smart door lock*, *Blynk*, dan *face recognition* sebagai salah satu fitur keamanan sebuah bangunan. *Smart door lock* memungkinkan pemilik bangunan mengendalikan akses pintu dari jarak jauh melalui *smartphone*. Penerapan teknologi *IoT* melalui *Blynk* memfasilitasi pengelolaan antarmuka kendali secara efisien karena dapat menghubungkan perangkat seperti *Arduino* dan *Raspberry Pi* ke jaringan *WiFi* sehingga pintu dapat dikendalikan dari mana saja. Dengan penambahan fitur *face recognition*, akses pintu menjadi lebih aman karena hanya pengguna yang terverifikasi melalui pengenalan wajah yang dapat membuka pintu.

Integrasi *smart door lock*, *Blynk*, dan *face recognition* merupakan salah satu pengembangan *IoT* dalam menjaga keamanan bangunan yang memungkinkan pengguna mengendalikan akses pintu sebuah bangunan dengan tingkat keamanan yang dikembangkan dengan *face recognition*. Sistem ini juga diharapkan dapat memudahkan pengguna dalam mengendalikan akses pintu tanpa batasan jarak melalui *Blynk* serta melakukan pengawasan terhadap pengendalian akses pintu. Oleh karena itu, penelitian berjudul "*Smart door lock* Sistem Berbasis Aplikasi *Blynk* dengan *Face Recognition* Menggunakan *CNN*" penting dilakukan.

## 2. METODOLOGI PENELITIAN

Dalam penelitian ini, pengembangan sistem kunci pintu pintar dilakukan menggunakan aplikasi *Blynk* yang terhubung dengan fitur pengenalan wajah berbasis metode *Convolutional Neural Network* (*CNN*). Metodologi yang digunakan meliputi tahapan desain perangkat keras dan perangkat lunak, integrasi aplikasi, serta penerapan *CNN* untuk mendeteksi dan mengenali wajah pengguna.

### 2.1. Jenis Penelitian

Penelitian ini merupakan penelitian eksperimental dengan metode *prototype*. Penelitian eksperimental dilakukan untuk menguji pengaruh dan performa dari sebuah *prototype* dalam kondisi yang terkontrol [17]. Penelitian bertujuan untuk mengembangkan dan menguji sistem *smart door lock* menggunakan aplikasi *Blynk* dengan *face recognition*.

## 2.2. Metode Pengumpulan Data

Penelitian ini menggunakan beberapa jenis metode penelitian, yaitu:

### 1) Observasi

Metode observasi digunakan untuk mengamati cara kerja dari *smart door lock* yang telah dibuat. Observasi ini dilakukan untuk memastikan bahwa sistem berjalan dengan baik dan sesuai dengan perencanaan.

### 2) Wawancara

Wawancara dilakukan baik dengan ahli untuk mendapatkan masukan dan saran mengenai sistem yang sedang dikembangkan. Selain itu, wawancara juga dilakukan dengan pengguna untuk mengetahui kebutuhan dan harapan mereka terhadap sistem *smart door lock* yang dikembangkan.

### 3) Studi Literatur

Metode studi literatur digunakan untuk mengumpulkan dan menganalisis informasi dari berbagai sumber yang relevan seperti buku, jurnal ilmiah, artikel, dan sumber-sumber lainnya. Studi literatur ini bertujuan untuk meningkatkan pemahaman peneliti mengenai teknologi *face recognition*, aplikasi *Blynk*, dan sistem keamanan *smart door lock*.

## 2.2. Metode Pengembangan Sistem

Menurut Roger S. Pressman (2002) pendekatan prototipe atau *prototyping paradigm* sangat cocok di gunakan untuk sistem atau perangkat lunak yang dibangun mengikuti kebutuhan pengguna. Metode ini sangat sesuai diterapkan dalam proses perancangan perangkat lunak yang akan dibangun. Tahapan-tahapan dalam model proses prototipe mewakili tahapan pengembangan desain perangkat lunak baru yang akan dibangun.

Pada *prototype* ada tahapan-tahapan yang dilakukan, yaitu diantaranya:

1. Pengumpulan kebutuhan
2. Membangun *prototyping*
3. Evaluasi *prototyping*
4. Mengkodekan sistem
5. Menguji sistem
6. Evaluasi sistem
7. Menggunakan sistem.



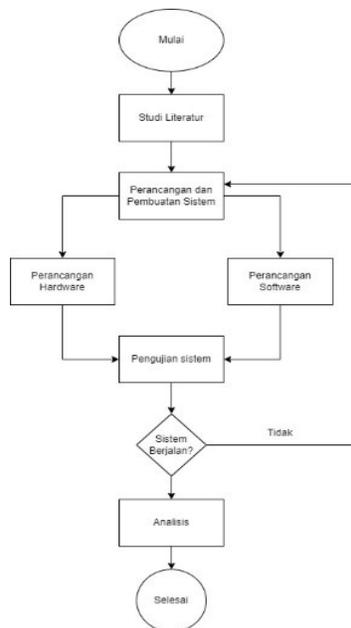
Gambar 1. Rencana Penelitian Sistem *Prototype*

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Implementasi Sistem

Rencana atau *planning* adalah sebuah langkah awal dalam sebuah penelitian untuk membuat rencana secara terstruktur yang merinci langkah-langkah untuk melakukan penelitian. Ini meliputi merumuskan tujuan penelitian, merancang metodologi, mengumpulkan dan menganalisis data, serta menginterpretasikan hasil penelitian. Rencana penelitian membantu memastikan bahwa penelitian dilakukan dengan terarah dan efisien, menghasilkan temuan yang bermakna, dan memberikan kontribusi pada pengetahuan dalam bidang yang diteliti.

Penelitian ini akan membuat sebuah sistem untuk membuat penguncian pintu otomatis secara *face recognition* dan aplikasi *Blynk*, dan juga *memonitoring* siapa saja yang akan masuk ke pintu dengan fitur memotret siapapun yang menekan *bell/button*. Untuk penjelasan rencananya dibuatlah sebuah diagram alur seperti berikut ini:



Gambar 2. Rencana Penelitian Sistem

#### 3.2. Rancangan Penelitian

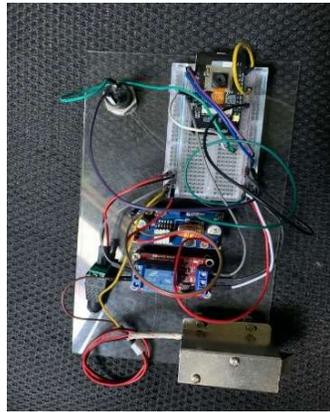
Penelitian ini bertujuan untuk membuat sebuah sistem *smart door lock* menggunakan *face recognition* dan juga aplikasi *Blynk*, juga dapat mengawasi setiap orang yang menekan tombol *bell/button* yang disediakan secara *realtime* melalui aplikasi *Blynk*.

Perancangan ini dibagi menjadi 2 bagian utama, yaitu:

1. Perancangan Perangkat keras (*Hardware*)
2. Perancangan perangkat lunak (*Software*).

#### 3.3. Desain Rancangan

Desain rancangan *smart lock* meliputi gambaran dari bagaimana alat ini akan diterapkan dalam instalasi rumah sebenarnya. Tampilan alat ini cukup sederhana dalam perancangannya, seperti yang dapat dilihat di bawah ini:

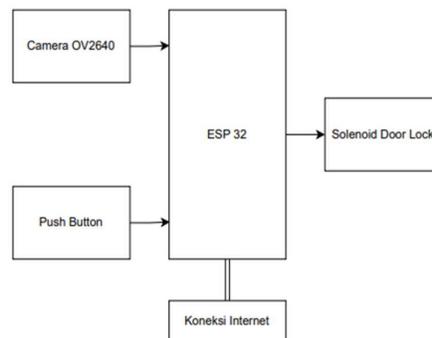


Gambar 3. Tampilan Hasil Rangkaian

### 3.4. Perancangan Diagram Blok Perangkat Keras

Perancangan diagram blok perangkat keras adalah representasi visual dari sistem atau proses yang kompleks, menggunakan blok-blok yang mewakili komponen atau elemen sistem tersebut. Setiap blok dalam diagram mewakili satu bagian dari sistem, dan hubungan antara blok-blok tersebut menggambarkan bagaimana komponen-komponen tersebut berinteraksi atau saling terhubung.

Dalam diagram blok, setiap blok biasanya diberi label dengan nama komponen atau fungsi yang diwakilinya. Garis atau panah digunakan untuk menghubungkan blok-blok tersebut, menunjukkan arah aliran informasi, energi, atau sinyal antara komponen-komponen tersebut. Beberapa blok juga dapat memiliki *input* dan *output* yang menunjukkan masukan dan keluaran dari blok tersebut.



Gambar 4. Blok Perangkat Keras

Fungsi dari setiap blok adalah:

#### 1. Blok *Input*

Kamera OV2640 berfungsi untuk merekam dan menyimpan setiap foto yang diinginkan oleh pengguna sebagai tamu yang diizinkan masuk tanpa harus izin terlebih dahulu, sedangkan *push button* berfungsi untuk melakukan foto kepada orang yang telah menekan *button* tersebut sehingga foto tersebut akan dikirim ke *Blynk*.



## 2. Blok Proses

Mikrokontroler yang digunakan di penelitian ini adalah ESP32, komponen ini berfungsi untuk memproses semua alat yang terpasang di perangkat ESP32 cam itu sendiri. Disini alat yang digunakan adalah *Camera OV2640* dan juga *Push Button*.

Kamera berfungsi untuk mencari kecocokan data wajah yang ada di ESP32 cam dengan tamu yang akan memasuki rumah, jika kecocokan data dan juga tamu sudah cocok, maka otomatis ESP32 akan mengirimkan perintah kepada *relay* untuk menyalakan *solenoid door lock* agar kunci terbuka, akan tetapi jika wajah tidak sesuai dengan *database* yang ada maka kunci tidak akan terbuka.

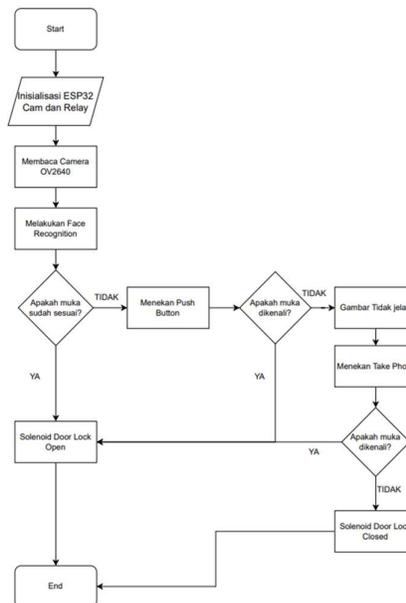
*Push button* berfungsi untuk mengirimkan perintah untuk melakukan *capture* foto ke kamera, yang mana ketika foto berhasil diambil, foto tersebut akan dikirimkan ke *Blynk* untuk ditampilkan ke pengguna. Jika pengguna mengetahui siapa tamu tersebut, maka pengguna dapat membuka kunci secara manual melalui aplikasi *Blynk*, akan tetapi jika foto tidak bagus dan kurang jelas, maka pengguna pun dapat melakukan foto ulang dengan menekan tombol *take photo* yang ada di *Blynk*.

## 3. Blok Output

*Solenoid door lock* berfungsi untuk membuka dan menutup pintu secara otomatis dan manual sesuai perintah yang diberikan oleh ESP32.

### 3.5. Flowchart Sistem

Berikut *flowchart* dari sistem *Smart door lock* menggunakan aplikasi *Blynk* dengan *face recognition*:



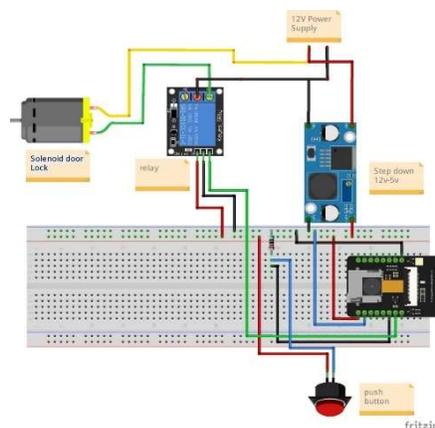
Gambar 5. Flowchart Sistem

Penjelasan *flowchart* dari *Smart door lock* menggunakan aplikasi *Blynk* dengan *face recognition* adalah sebagai berikut:



Diagram *flowchart* diawali dengan inisialisasi nilai awal dan konfigurasi mikrokontroler dilanjutkan membaca kamera. Ketika kamera melakukan deteksi wajah, maka di *web base* ESP32 cam akan muncul *notif* bahwa wajah tersebut diizinkan atau tidak. Jika wajah diizinkan maka pintu akan terbuka, tetapi jika wajah tidak diizinkan, maka pintu akan tetap tertutup. Kemudian tamu akan menekan tombol/*bell*, kemudian ESP32 cam akan melakukan pemotretan secara otomatis, dan foto tersebut akan ditampilkan di aplikasi *Blynk*. Kemudian pengguna harus melakukan pengecekan secara manual, jika pengguna mengenali tamu tersebut, maka pintu bisa dibuka secara manual menggunakan *Blynk*, akan tetapi jika foto tersebut tidak jelas, pengguna bisa melakukan foto ulang dengan menekan *take photo* di aplikasi *Blynk*. Jika pengguna tidak mengenali tamu tersebut, maka pintu tetap tertutup.

### 3.6. Rangkaian Sistem



Gambar 6. Rangkaian Sistem

Pembuatan rancangan sistem *Smart door lock* ini menggunakan ESP32 sebagai mikrokontroler untuk mengontrol semua alat yang ada. Adapun beberapa alat yang digunakan yaitu sensor *Camera OV2640* dan *Relay*. *Camera OV2640* sebagai pendeteksi wajah dan foto tamu yang menekan *button/bell*, dan *relay* untuk mengatur buka tutup kunci. Semua alat tersebut dipadukan dan dikendalikan oleh ESP32 lalu dijadikan *output* ke aplikasi android menggunakan *Blynk*. Kemudian *coding* diperlukan untuk memadukan semua perangkat keras tersebut agar berfungsi sebagai *Smart door lock*, nantinya setiap foto yang diambil oleh kamera akan ditampilkan ke aplikasi *Blynk*.

### 3.7. Coding Program

*Blynk auth* token adalah token yang didapat dari *Blynk* aplikasinya, bisa juga di copy langsung dari *Blynk* nya. Fungsinya untuk mengetahui token *Blynk* mana yang akan dihubungkan dengan alat.

```
10 #define BLYNK_TEMPLATE_ID "TMPL6BApzMU-i"  
11 #define BLYNK_TEMPLATE_NAME "ESP32 Cam"  
12 #define BLYNK_AUTH_TOKEN "brdfEvaCGP9pfxb5GgW5lY7AaEbxE3P8"
```

Gambar 7. Tampilan *Blynk Auth* Token



Jika sudah ada token dan alat sudah ada *Wifi*, berikutnya dihubungkan dengan *code Blynk*, seperti ini:

```
153 Serial.println( "OLW KONEK Blynk ");  
154 Blynk.begin(BLYNK_AUTH_TOKEN, ssid, password);  
155 // Blynk.begin(BLYNK_AUTH_TOKEN, ssid, password);
```

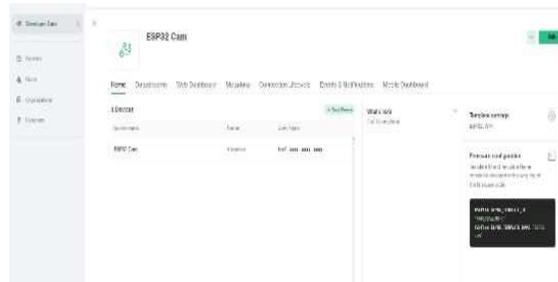
Gambar 8. Tampilan Code Blynk

Jika sudah, nyalakan fitur *Blynk* dengan *Blynk.run* pada *void loop*.

```
177 Blynk.run();
```

Gambar 9. Tampilan Run Pada Void Loop

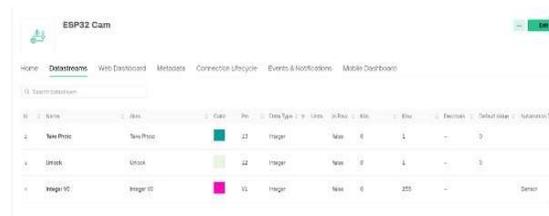
Untuk mengetahui apakah data sudah siap atau belum, *login* ke *Blynk* lalu masuk ke *developer zone* ada di kanan bawah. *Auth token* berada di sebelah tab status.



Gambar 10. Tampilan Blynk Developer Zone

Kemudian untuk *data stream* terdapat tiga, yaitu:

1. *Take Photo*
2. *Unlock*
3. *Integer*



Gambar 11. Tampilan Blynk Data stream

Tampilan akan seperti ini, untuk *take photo* dengan catatan alat dengan *smartphone* harus 1 jaringan yaitu 1 *Wifi*.



Gambar 12. Tampilan *Blynk Take Photo*

*Integer* memiliki *trigger* pada *void take photo*, line 56, di bagian *Blynk.setProperty*, guna untuk mengirim hasil gambar ke *Blynk*.

```
50 void takePhoto()
51 {
52   digitalWrite(LED, HIGH);
53   delay(200);
54   uint32_t randomNum = random(50000);
55   Serial.println("http://"+local_IP+"/capture?cb="+ (String)randomNum);
56   Blynk.setProperty(V1, "urls", "http://"+local_IP+"/capture?cb="+ (String)randomNum);
57   digitalWrite(LED, LOW);
58   delay(1000);
59 }
```

Gambar 13. Tampilan *Blynk Void Take Photo*.

Kode untuk tombol '*Unlock*' di *Blynk* tidak disertakan di sini karena aplikasi ini membaca langsung pin dari ESP32-CAM. Tidak ada kode *Blynk* yang murni, hanya menggunakan *digital Write* dengan syarat bahwa variabel '*Unlock*' harus dibaca terlebih dahulu saat posisinya sudah *HIGH* (1). Ketika tombol *unlock* ditekan, itu akan menjadi *high* (1).

```
if(digitalRead(unlock) == HIGH && !relayStateButton){
  relayStateButton = true;
  digitalWrite(relayPin,LOW);
  Serial.println("Pintu terbuka via blynk");
  startTimeButton = millis();
  // Serial.println(relayState);
  // delay(5000);
  // digitalWrite(relayPin,LOW);
}
```

Gambar 14. Tampilan *High*

Begitupun sama halnya dengan *Take Photo* di *Blynk*.

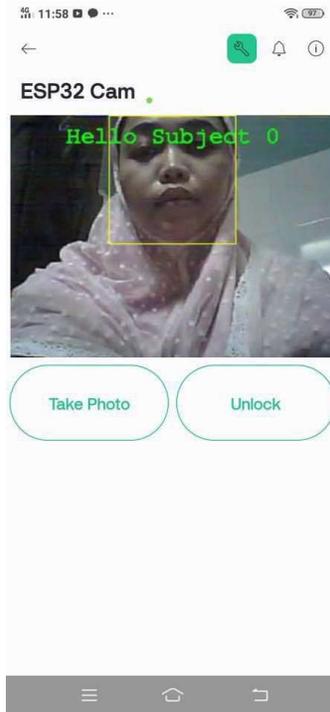
```
if (digitalRead(blynk_button) == 1 && takePhotoButtonBlynkState == 0){
  takePhotoButtonBlynkState = 1;
  takePhoto();
  Serial.println("Foto diambil dari blynk");
  // delay(3000);
}
```

Gambar 15. Tampilan *Take Photo*

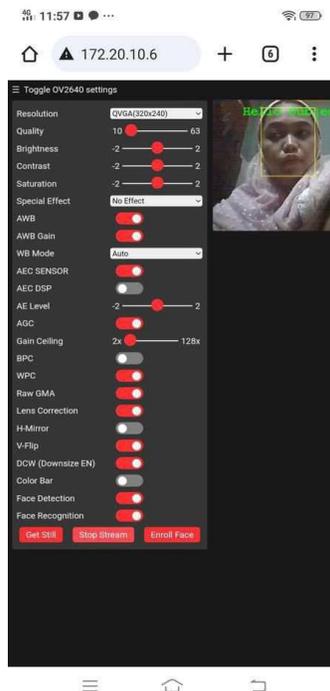


### 3.8. Pengujian

Pengujian dilakukan dengan menguji setiap perangkat keras (*Hardware*) yang terpasang. Jika setiap fungsi pada perangkat berfungsi dengan baik dan tidak ada masalah, maka perangkat tersebut akan diintegrasikan menjadi sebuah *prototype*.



Gambar 16. Tampilan Pengujian pada Aplikasi



Gambar 17. Tampilan Pengujian Pada *Face Recognition*.



#### 4. KESIMPULAN

Penelitian dan pengembangan *Smart door lock* berbasis aplikasi *Blynk* dengan *face recognition* menggunakan metode *Convolutional Neural Network* (CNN) telah berhasil dilakukan. Sistem ini mampu meningkatkan keamanan rumah dengan mengenali wajah berdasarkan kategori "dikenali" dan "tidak dikenali." Penerapan CNN terbukti efektif dalam pengenalan wajah, memberikan tingkat akurasi tinggi, dan menghadirkan pengalaman pengguna yang modern serta terhubung secara teknologi.

#### 5. SARAN

Pengembangan sistem sebaiknya difokuskan pada pembaruan *dataset* wajah yang lebih representatif untuk memastikan akurasi pengenalan dalam berbagai kondisi pencahayaan dan ekspresi wajah. Selain itu, integrasi dengan teknologi keamanan tambahan seperti enkripsi data, deteksi gerakan, atau notifikasi keamanan dapat meningkatkan perlindungan sistem. Memastikan stabilitas operasional dan kemampuan akses jarak jauh melalui aplikasi *Blynk* juga menjadi langkah penting untuk memberikan pengalaman pengguna yang optimal.

#### DAFTAR PUSTAKA

- [1] A. Fakhruddin, "Rancang Bangun Sistem Keamanan Pintu Rumah Berbasis Internet of Things Dengan Esp32 dan Aplikasi Blynk," *J. Tek. Elektro dan Inform.*, vol. 19, no. 1, pp. 53–59, 2024, [Online]. Available: [file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/document \(2\).pdf](file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/document (2).pdf)
- [2] D. Andriyan, R. A. Yusda, and M. D. Sena, "Rancang Bangun Smart Door Lock Berbasis IoT Untuk Smart Office," *JUTSI J. Teknol. dan Sist. Inf.*, vol. 4, no. 1, pp. 1–10, 2024, [Online]. Available: <file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/3004-8777-2-PB.pdf>
- [3] A. S. Bakhri, K. Suhada, and K. Kamaludin, "Perancangan Sistem Doorlock Menggunakan Aplikasi Blynk Berbasis IoT Studi Kasus Pada Rumah Tempat Tinggal Pribadi," *Pros. Semin. Nas. Inov. dan Adopsi Teknol.*, vol. 1, no. 1, pp. 1–10, 2021, [Online]. Available: <https://ejournal.rosma.ac.id/index.php/inotek/article/view/161>
- [4] N. Soedjarwanto, "Prototipe Smart Dor Lock Menggunakan Motor Stepper Berbasis Iot (Internet Of Things)," *Electrician*, vol. 15, no. 2, pp. 73–82, 2021, doi: 10.23960/elc.v15n2.2167.
- [5] A. N. Faishal, Y. Heriadi, and Annah, "Implementasi Google Voice Pada Sistem Smart Door Lock Berbasis Arduino," *Tek. Inform.*, vol. 481, no. 2, pp. 481–489, 2023, [Online]. Available: <file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/1503-Article Text-2446-2-10-20240319.pdf>
- [6] J. Ambarita, R. A. P, and A. S. Wibowo, "Rancang Bangun Prototipe Smarhome Berbasis Internet of Things (IoT) Menggunakan Aplikasi Blynk Dengan Modul ESP 8266," *e-Proceeding Eng.*, vol. 6, no. 2, pp. 3006–3013, 2019, [Online]. Available: <file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/10502-Article Text-20399-1-10-20191126.pdf>
- [7] T. Juwariyah, S. Prayitno, and A. Mardhiyah, "Perancangan Sistem Deteksi Dini Pencegah Kebakaran Rumah Berbasis IoT (Internet of Things)," *Semin. Nas. Inform. Sist. Inf. Dan Keamanan Siber*, vol. 1, no. 1, pp. 57–62, 2018, [Online]. Available: <file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/36-118-1-PB.pdf>



- [8] R. Berlianti and Fibriyanti, "Perancangan Alat Pengontrolan Beban Listrik Satu Phasa Jarak Jauh Menggunakan Aplikasi Blynk Berbasis Arduino Mega," *Sain, Energi Teknol. Ind.*, vol. 5, no. 1, pp. 17–26, 2020, [Online]. Available: file:///D:/PUNYA AFI/Portofolio/Skripsi Retno/6398-Article Text-19897-1-10-20210405.pdf
- [9] E. Fadly, S. Adi Wibowo, and A. Panji Sasmito, "Sistem Keamanan Pintu Kamar Kos Menggunakan Face Recognition Dengan Telegram Sebagai Media Monitoring Dan Controlling," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 5, no. 2, pp. 435–442, 2021, doi: 10.36040/jati.v5i2.3796.