



Lampiran 1. Kode Program ESP32_AuthenticationUnit

```

void scanFingerprint() {
    if (waitingForFace) return;
    if (currentEnrollState != STATE_IDLE) return;
    uint8_t p = finger.getImage();
    if (p == FINGERPRINT_NOFINGER) return;
    if (p != FINGERPRINT_OK) return;
    p = finger.image2Tz();
    if (p != FINGERPRINT_OK) return;
    p = finger.fingerSearch();
    if (p == FINGERPRINT_OK) {
        uint8_t id = finger.fingerID;
        String name = readNameFromEEPROM(id);
        String uid = readUIDFromEEPROM(id);
        Serial.printf("[FP] Cocok! ID=%d Nama=%s\n", id, name.c_str());

        // sesi 2FA
        sessionAuthID = id;
        sessionNonce = generateNonce();
        sessionStartTime = millis();
        sessionMethod = "Fingerprint";
        sessionUID = "";
        waitingForFace = true;

        DynamicJsonDocument doc(256);
        doc["auth_id"] = sessionAuthID;
        doc["nonce"] = sessionNonce;

        String payload;
        serializeJson(doc, payload);
        mqttClient.publish(topic_stream_cam, payload.c_str());
        ledSuccessful();
        Serial.println("[2FA] Challenge dikirim: " + payload);
        uiSet("VERIFIKASI WAJAH", "via " + cameraIP, "dan Arahkan Wajah", "ke Kamera",
            UI_FACE_WAIT, 5000);
    } else {
        Serial.println("[FP] Tidak terdaftar");
        ledFailed();
        uiSet("AKSES DITOLAK", "", "Sidik Jari Anda", "Tidak Terdaftar", UI_DENIED, 3500);
        sessionMethod = "Fingerprint";
        sessionUID = "";

        if (WiFi.status() == WL_CONNECTED)
            fbLog(0, "-", "", "Fingerprint", "auth_failed");
        else
            Serial.println("[FB] Offline, skipping denied log.");
    }
}

void scanRFID() {
    if (waitingForFace) return;
    if (currentEnrollState != STATE_IDLE) return;
    if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial()) return;

    String uid = "";
    for (byte i = 0; i < rfid.uid.size; i++) {

```

```

    if (rfid.uid.uidByte[i] < 0x10) uid += "0";
    uid += String(rfid.uid.uidByte[i], HEX);
  }
  uid.toUpperCase();
  currentEnrollUID = uid;
  rfid.PICC_HaltA();
  rfid.PCD_StopCrypto1();

  for (uint8_t i = 1; i <= MAX_ID; i++) {
    if (readUIDFromEEPROM(i).equalsIgnoreCase(uid)) {
      String name = readNameFromEEPROM(i);
      Serial.printf("[RFID] Cocok! ID=%d Nama=%s\n", i, name.c_str());

      // sesi 2FA
      sessionAuthID = i;
      sessionNonce = generateNonce();
      sessionStartTime = millis();
      sessionMethod = "RFID";
      sessionUID = uid;
      waitingForFace = true;

      DynamicJsonDocument doc(256);
      doc["auth_id"] = sessionAuthID;
      doc["nonce"] = sessionNonce;

      String payload;
      serializeJson(doc, payload);

      mqttClient.publish(topic_stream_cam, payload.c_str());
      ledSuccessful();
      Serial.println("[2FA] Challenge dikirim: " + payload);
      uiSet("VERIFIKASI WAJAH", "Via " + cameraIP, "dan Arahkan Wajah", "ke Kamera",
        UI_FACE_WAIT, 5000);
      return;
    }
  }
  Serial.println("[RFID] Tidak terdaftar");
  ledFailed();
  uiSet("AKSES DITOLAK", "", "RFID anda", "Tidak Terdaftar", UI_DENIED, 3500);
  sessionMethod = "RFID";
  sessionUID = "-";

  if (WiFi.status() == WL_CONNECTED)
    fbLog(0, "-", "-", "RFID", "auth_failed");
  else
    Serial.println("[FB] Offline, skipping denied log.");
}

```

Kode program selengkapnya dapat dilihat pada repositori GitHub berikut:

https://github.com/waradiana/SmartLockSecuritySystem/tree/main/ESP32_AuthenticationUnit

Lampiran 2. Kode Program ESP32_DoorLockUnit

```

void loop() {
    handleWiFiReset();

    if (!wifiPortalActive) {
        wifiConnect();
    }

    // MQTT reconnect & loop
    if (WiFi.status() == WL_CONNECTED) {
        mqttReconnectNonBlocking();
        if (mqttClient.connected()) {
            mqttReady = true;
            mqttClient.loop();
        } else {
            mqttReady = false;
        }
    } else {
        mqttReady = false;
    }

    // OLED + WiFi state
    if (WiFi.status() == WL_CONNECTED) {
        wifiWasConnected = true;

        if (!mqttReady) {
            uiSet("WIFI CONNECTED", "Menunggu MQTT...");
        } else if (!relayActive) {
            uiSet("SYSTEM READY", "PUSH TO EXIT");
        }
    } else {
        if (wifiWasConnected && !wifiPortalActive) {
            startReconnectPortal();
        }
    }

    // Relay Timer
    if (relayActive && millis() - relayStartTime >= relayDuration) {
        digitalWrite(RELAY_PIN, LOW);
        relayActive = false;
        Serial.println("[Relay] Auto OFF");
        uiSet("SYSTEM READY", "PUSH TO EXIT");
    }

    // Door Switch Sensor
    int doorState = digitalRead(DOOR_SENSOR_PIN);
    if (doorState != lastDoorState && millis() - lastDoorReport > 100) {
        lastDoorState = doorState;
        lastDoorReport = millis();

        if (doorState == HIGH) {
            Serial.println("Door Open");
            mqttClient.publish(topic_door_status, "open");
        } else {
            Serial.println("Door Closed");
            mqttClient.publish(topic_door_status, "closed");
        }
    }
}

```



```
}  
  
// Button Exit  
if (digitalRead(EXIT_BTN) == LOW) {  
  if (millis() - lastButtonPress > 300) {  
    Serial.println("[Button] Manual OPEN");  
    mqttClient.publish(topic_exit_button, "exit");  
    digitalWrite(RELAY_PIN, HIGH);  
    relayActive = true;  
    relayStartTime = millis();  
    uiSet("DOOR EXIT", "Pintu Terbuka");  
    lastButtonPress = millis();  
  }  
}  
}  
shortDelay(5);  
}
```

Kode program selengkapnya dapat dilihat pada repositori GitHub berikut:

https://github.com/waradiana/SmartLockSecuritySystem/tree/main/ESP32_DoorLockUnit



Lampiran 3. Kode Program ESP32Cam_FaceRecognitionUnit

```

void reconnectMQTTNonBlocking() {
    if (mqttClient.connected()) return;
    if (millis() - lastMqttReconnectAttempt < mqttRetryInterval) return;
    lastMqttReconnectAttempt = millis();

    String clientId = "ESP32_CamUnit";
    clientId += String((uint32_t)ESP.getEfuseMac(), HEX);

    Serial.print("[MQTT] Menghubungkan ke HiveMQ Cloud...");
    if (mqttClient.connect(clientId.c_str(), mqtt_user, mqtt_pass)) {
        Serial.println("Terhubung ✅");

        mqttClient.subscribe(topic_stream_cam);
        mqttClient.subscribe(topic_enroll_face);
        mqttClient.subscribe(topic_delete_face);
        mqttClient.subscribe(topic_reset_WiFi);

        mqttClient.setCallback(mqtt_callback);

        publishCameraIP();
    } else {
        Serial.print("Gagal, rc=");
        Serial.println(mqttClient.state());
    }
}

// MQTT Callback Publish & Subscribe
void mqtt_callback(char* topic, byte* payload, unsigned int length) {
    String message = "";
    for (unsigned int i = 0; i < length; i++) message += (char)payload[i];
    message.trim();

    Serial.print("[MQTT] ");
    Serial.println(topic);
    Serial.print("=> ");
    Serial.println(message);

    // Trigger start stream
    if (String(topic) == topic_stream_cam) {
        DynamicJsonDocument doc(256);
        DeserializationError error = deserializeJson(doc, message);
        if(error){
            Serial.println("[2FA] JSON tidak valid");
            return;
        }

        activeAuthID = doc["auth_id"];
        activeNonce = doc["nonce"].as<String>();

        sessionActive = true;
        streamTrigger = true;
        sessionStartMs = millis();

        lastFaceState = false;
        matchFaceId = 0;
        publishedThisSession = false;
    }
}

```

```

Serial.println("[2FA] Session 2FA dimulai");
return;
}

// Trigger mode enroll face
if (String(topic) == topic_enroll_face && message == "true") {
    streamTrigger = true;
    enrollTrigger = true;
    Serial.println("[ENROLL] Mode enroll wajah aktif");
    return;
}

// Hapus face ID dari SD
if (String(topic) == topic_delete_face) {
    if (message.startsWith("delete ")) {
        String idPart = message.substring(7);
        int idToDelete = idPart.toInt();

        if (idToDelete > 0) {
            char path[32];
            sprintf(path, "/face_%d.bin", idToDelete);

            if (SD_MMC.exists(path)) {
                SD_MMC.remove(path);
                Serial.printf("MQTT: Face ID %d dihapus dari SD\n", idToDelete);
                clear_faces_in_ram();
                load_faces_from_sd();
            }
            else {
                Serial.println("ID tidak ditemukan di SD");
            }
        }
    }
}

// Reset WiFi
if (String(topic) == topic_reset_wifi) {
    if (message == "reset_wifi") {
        Serial.print("⚠ Perintah Reset WiFi Diterima");

        WiFiManager wm;
        wm.resetSettings();

        Serial.println("Konfigurasi WiFi di Hapus. Restarting...");
        delay(1500);
        ESP.restart();
        return;
    }
}

// MQTT Publish Info User Status & ID
void publishInfoUser(int userId) {

    if(!sessionActive) return;

    DynamicJsonDocument doc(256);

```

```

doc["face_id"] = userId;
doc["nonce"] = activeNonce;

String output;
serializeJson(doc, output);

mqttClient.publish(topic_face_id, output.c_str());
Serial.println("[2FA] Face result dikirim: " + output);

sessionActive = false;
streamTrigger = false;
}

// MQTT Publish Info Enroll URL
void publishCameraIP() {
  String ip = getLocalIpString();
  mqttClient.publish(topic_ip_cam, ip.c_str(), true); // retained supaya Auth langsung dapat IP terakhir
  Serial.println("[MQTT] Camera IP dikirim: " + ip);
}

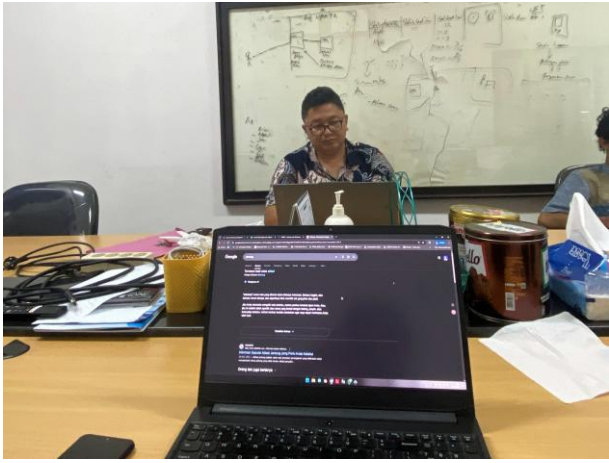
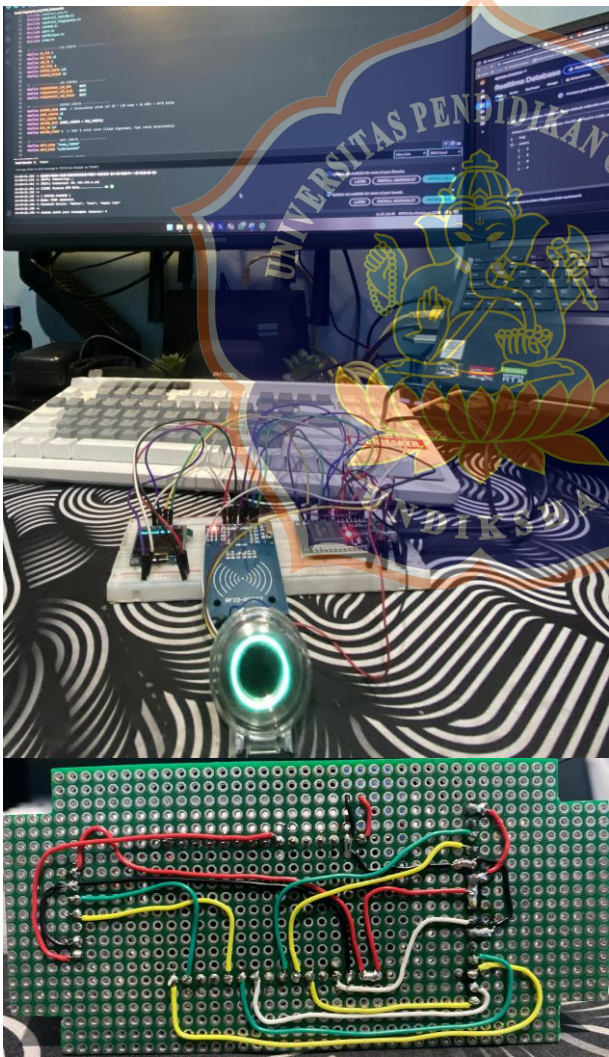
```

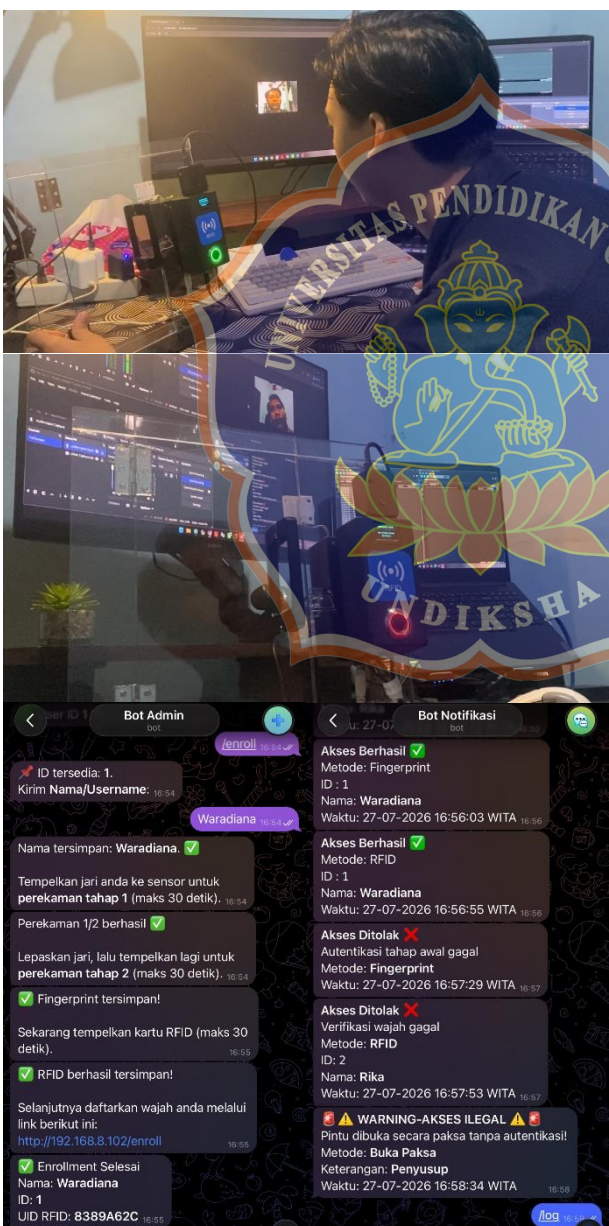
Kode program selengkapnya dapat dilihat pada repositori GitHub berikut:

https://github.com/waradiana/SmartLockSecuritySystem/tree/main/ESP32Cam_FaceRecognitionUnit



Lampiran 4. Dokumentasi Penelitian

Gambar	Keterangan
	Mengidentifikasi permasalahan yang timbul pada sistem penguncian untuk akses ke ruang server bersama Kepala UPA TIK Undiksha.
	Proses perakitan, pemrograman, dan integrasi <i>Smart Lock System</i>.

Gambar	Keterangan
	
	Pengujian terhadap fungsional fitur dan performa pada sistem autentikasi pada perangkat <i>Smart Lock System</i> secara mandiri.

Gambar	Keterangan
	Evaluasi pengalaman penggunaan prototipe perangkat <i>Smart Lock System</i>, serta pengisian kuesioner dengan responden dari pihak Staf UPA TIK Undiksha secara luring.

Lampiran 5. Hasil Kuesioner *Usability Testing*

LEMBAR KUESIONER *USABILITY TESTING*

Dengan hormat, dimohon kiranya Bapak/Ibu bersedia mengisi kuesioner ini dengan melengkapi data diri sebagai berikut.

No. Responden : 1
 Nama Responden : *Khot Nawar Arimbawa*
 Jabatan : *Staf UPA TIK*
 Hari/Tanggal : *Jlala, 28-07-2016*

Berilah tanda (✓) pada kolom Skor (1-5) yang paling sesuai dengan persepsi Anda setelah mencoba sistem di atas.

Keterangan Skala:

1 = Sangat Tidak Setuju

2 = Tidak Setuju

3 = Netral

4 = Setuju

5 = Sangat Setuju

No.	Pertanyaan	Skor				
		1	2	3	4	5
1.	Saya rasa saya akan sering menggunakan sistem penguncian ini untuk operasional ruang server.					✓
2.	Saya merasa sistem penguncian ini terlalu rumit untuk hal yang sederhana.		✓			
3.	Saya merasa sistem penguncian ini sangat mudah digunakan.					✓
4.	Saya rasa saya memerlukan bantuan orang lain/teknisi khusus untuk bisa menggunakan sistem penguncian ini.	✓				
5.	Saya merasa fitur-fitur pada sistem penguncian ini terintegrasi dengan sangat baik.					✓
6.	Saya merasa banyak hal yang tidak konsisten (membingungkan) pada sistem penguncian ini.		✓			
7.	Saya rasa orang lain (staff/anggota lain) akan cepat belakar menggunakan sistem penguncian ini.				✓	
8.	Saya merasa sistem penguncian ini sangat repot/tidak praktis saat digunakan.	✓				
9.	Saya merasa sangat percaya diri saat mengoperasikan sistem penguncian ini.					✓
10.	Saya harus mempelajari banyak hal terlebih dahulu sebelum bisa menggunakan sistem penguncian ini.		✓			

Saran dan Masukan Tambahan:

--



LEMBAR KUESIONER *USABILITY TESTING*

Dengan hormat, dimohon kiranya Bapak/Ibu bersedia mengisi kuesioner ini dengan melengkapi data diri sebagai berikut.

No. Responden : 2
 Nama Responden : Made Agus Pangi Syaya
 Jabatan : Penata Layanan Operasional
 Hari/Tanggal : Selasa, 28 Juli 2026

Berilah tanda (✓) pada kolom Skor (1-5) yang paling sesuai dengan persepsi Anda setelah mencoba sistem di atas.

Keterangan Skala:

- 1 = Sangat Tidak Setuju
 2 = Tidak Setuju
 3 = Netral
 4 = Setuju
 5 = Sangat Setuju

No.	Pertanyaan	Skor				
		1	2	3	4	5
1.	Saya rasa saya akan sering menggunakan sistem penguncian ini untuk operasional ruang server.				✓	
2.	Saya merasa sistem penguncian ini terlalu rumit untuk hal yang sederhana.			✓		
3.	Saya merasa sistem penguncian ini sangat mudah digunakan.				✓	
4.	Saya rasa saya memerlukan bantuan orang lain/teknisi khusus untuk bisa menggunakan sistem penguncian ini.			✓		
5.	Saya merasa fitur-fitur pada sistem penguncian ini terintegrasi dengan sangat baik.					✓
6.	Saya merasa banyak hal yang tidak konsisten (membingungkan) pada sistem penguncian ini.		✓			
7.	Saya rasa orang lain (staff/anggota lain) akan cepat belakar menggunakan sistem penguncian ini.					✓
8.	Saya merasa sistem penguncian ini sangat repot/tidak praktis saat digunakan.	✓				
9.	Saya merasa sangat percaya diri saat mengoperasikan sistem penguncian ini.					✓
10.	Saya harus mempelajari banyak hal terlebih dahulu sebelum bisa menggunakan sistem penguncian ini.					✓

Saran dan Masukan Tambahan:

Tombol reset wifi di auth dan cam perlu di reposisi
agar tdk ada kemungkinan orang luar sembarangan
reset sistem

Singaraja, 28 Juli 2026

Responden.



LEMBAR KUESIONER *USABILITY TESTING*

Dengan hormat, dimohon kiranya Bapak/Ibu bersedia mengisi kuesioner ini dengan melengkapi data diri sebagai berikut.

No. Responden : 3
 Nama Responden : Made Yuda Sadewa, S.Pd., M.kom
 Jabatan : Pengelola layanan operasional
 Hari/Tanggal : Selasa, 28 Juli 2026

Berilah tanda (✓) pada kolom Skor (1-5) yang paling sesuai dengan persepsi Anda setelah mencoba sistem di atas.

Keterangan Skala:

1 = Sangat Tidak Setuju

2 = Tidak Setuju

3 = Netral

4 = Setuju

5 = Sangat Setuju

No.	Pertanyaan	Skor				
		1	2	3	4	5
1.	Saya rasa saya akan sering menggunakan sistem penguncian ini untuk operasional ruang server.			✓		
2.	Saya merasa sistem penguncian ini terlalu rumit untuk hal yang sederhana.	✓				
3.	Saya merasa sistem penguncian ini sangat mudah digunakan.					✓
4.	Saya rasa saya memerlukan bantuan orang lain/teknisi khusus untuk bisa menggunakan sistem penguncian ini.		✓			
5.	Saya merasa fitur-fitur pada sistem penguncian ini terintegrasi dengan sangat baik.				✓	
6.	Saya merasa banyak hal yang tidak konsisten (membingungkan) pada sistem penguncian ini.		✓			
7.	Saya rasa orang lain (staff/anggota lain) akan cepat belakar menggunakan sistem penguncian ini.					✓
8.	Saya merasa sistem penguncian ini sangat repot/tidak praktis saat digunakan.	✓				
9.	Saya merasa sangat percaya diri saat mengoperasikan sistem penguncian ini.				✓	
10.	Saya harus mempelajari banyak hal terlebih dahulu sebelum bisa menggunakan sistem penguncian ini.	✓				

Saran dan Masukan Tambahan:

- Pada saat mendaftarkan user baru pastikan tidak ada kendala seperti wajah yang terdeteksi sama.
- Pada saat hapus data, bisa ditambahkan/dimunculkan id user terlebih dulu, karena user mungkin lupa terhadap id-nya.

Singaraja, 28 Juli 2024.....

Responden.



RIWAYAT HIDUP



Made Waradiana Aryadi lahir di Singaraja pada tanggal 6 Juli 2003. Penulis lahir dari pasangan suami istri, bernama Ketut Utama Arya dan Ni Made Liesa Herliyani. Penulis berkebangsaan Indonesia dan beragama Hindu. Saat ini penulis beralamat di Jalan Jelantik Gingsir No. 123, Kecamatan Sukasada, Kabupaten Buleleng, Provinsi Bali.

Penulis menyelesaikan pendidikan dasar di SD Negeri 3 Sukasada dan lulus pada tahun 2015. Kemudian penulis melanjutkan pendidikan di SMP Negeri 1 Sukasada dan lulus pada tahun 2018. Pada tahun 2021, penulis lulus dari SMA Negeri 2 Singaraja jurusan Matematika dan Ilmu Pengetahuan Alam (MIPA) dan melanjutkan ke jenjang Sarjana Jurusan Teknik Informatika dengan mengambil Program Studi S1 Ilmu Komputer di Universitas Pendidikan Ganesha. Selanjutnya, mulai tahun 2021 hingga penulisan skripsi ini, penulis masih terdaftar sebagai mahasiswa Program Studi S1 Ilmu Komputer di Universitas Pendidikan Ganesha.

