

קבוצה 2 - דוח תרגיל סוחין

הדר בן עמי 211329461

מור בגין 207175977

אלמוג פישמן 208135806

רומי בן שחר 209521707

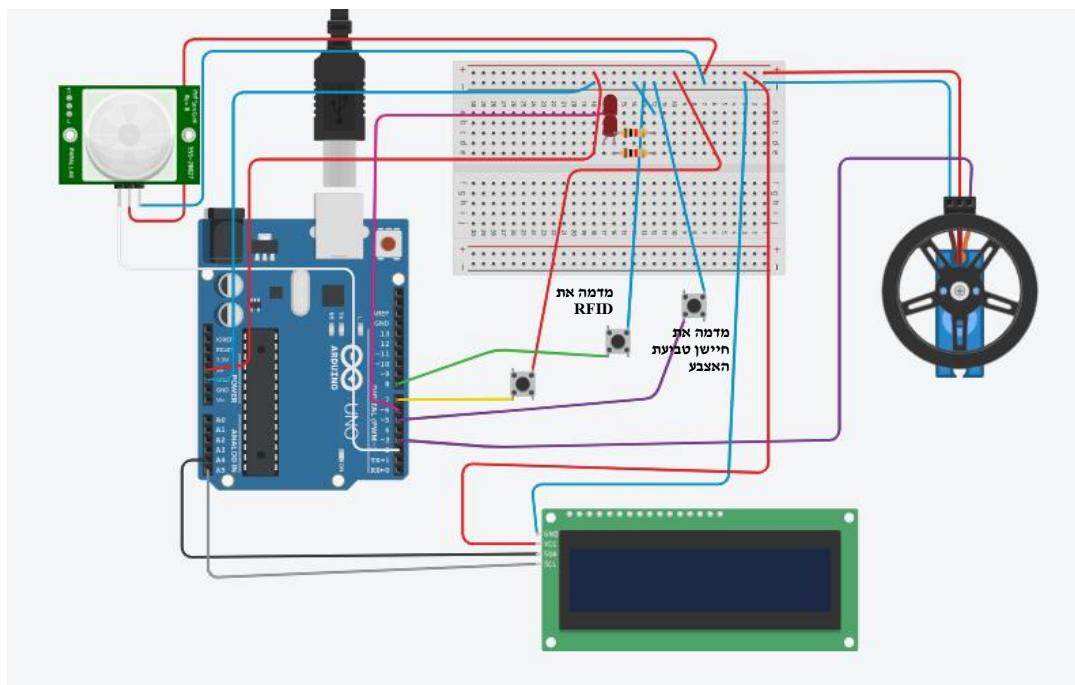
תיאור מטרת המערכת: הפרויקט מציג אב-טיפוס למערכת בקרה המשמשת כ"חדר בקרה" טכני עבור מופע הפסטיגל. מטרת המערכת היא לאפשר לבמאי המופיע שליטה מבודדת ומואבטחת על אלמנטים מכניים ותאורה על הבמה. המערכת מודדת שرك הבמאי (או מפעיל מורשה) יכול להפעיל את הפעולים, ומספקת חיזוי בזמן אמיתי על מצב המערכת ועל סיום הפעולות בצורה בטוחה.

סיפור המסתגרת: התרחש מתמקד בבמאי הפסטיגל הנמצא בחדר הבקרה מאחורי הקלעים. כדי להתחילה רצף פעולות במופע (כמו הנפת שחקן או סיבוב תפארה), הבמאי צריך לעبور תהליך זיהוי ולאחר מכן להפעיל את המערכת בצורה ידנית. המערכת מדמה את ניהול המעבר בין שלב ההכנות בחדר הבקרה לבין ביצוע הפעולות על הבמה מול הקהל.

תיאור משימת הבקרה: תהליך העבודה של המערכת מתבצע לפי השלבים הבאים:

- אבטחת גישה לחדר הבקרה: כדי למנוע הפעלה בשוגג, הבמאי חייב להזדהות מול המערכת באמצעות חיישן טביעה אצבע. כגביי למקראה חירום, ניתן להשתמש בחישן RFID לפתיחת הדלת. עם הזיהוי, מסך LCD מציג הודעה ברוך הבא אישית.
- הפעלת האלמנט המכני: לאחר הזיהוי, הבמאי לחץ על כפתור הפעלה. לחיצה זו מפעילה את מנוע הסרוויס (המודמה במה מסתובבת) למשך זמן קצר של 5 שניות.
- ניטור סיום הפעולה: חיישן תנועה ממוקם בנקודת הסיום על הבמה. כאשר הוא מזהה שהתנועה הסת衣ימה (ירידה של השחקן מהבמה), הוא מפעיל את הלוד כסימן להצלחת המשימה ומעבר לשלב הבא שהוא הבהוב אורות וליזרים כחלק מהמשך המופע. לבסוף לאחר 5 שניות נוספות מפסיק את המערכת להפעלה הבאה.

להלן צילום מסך מתוכנת סימולטור TINKCARD המדמה את המערכת שלנו:



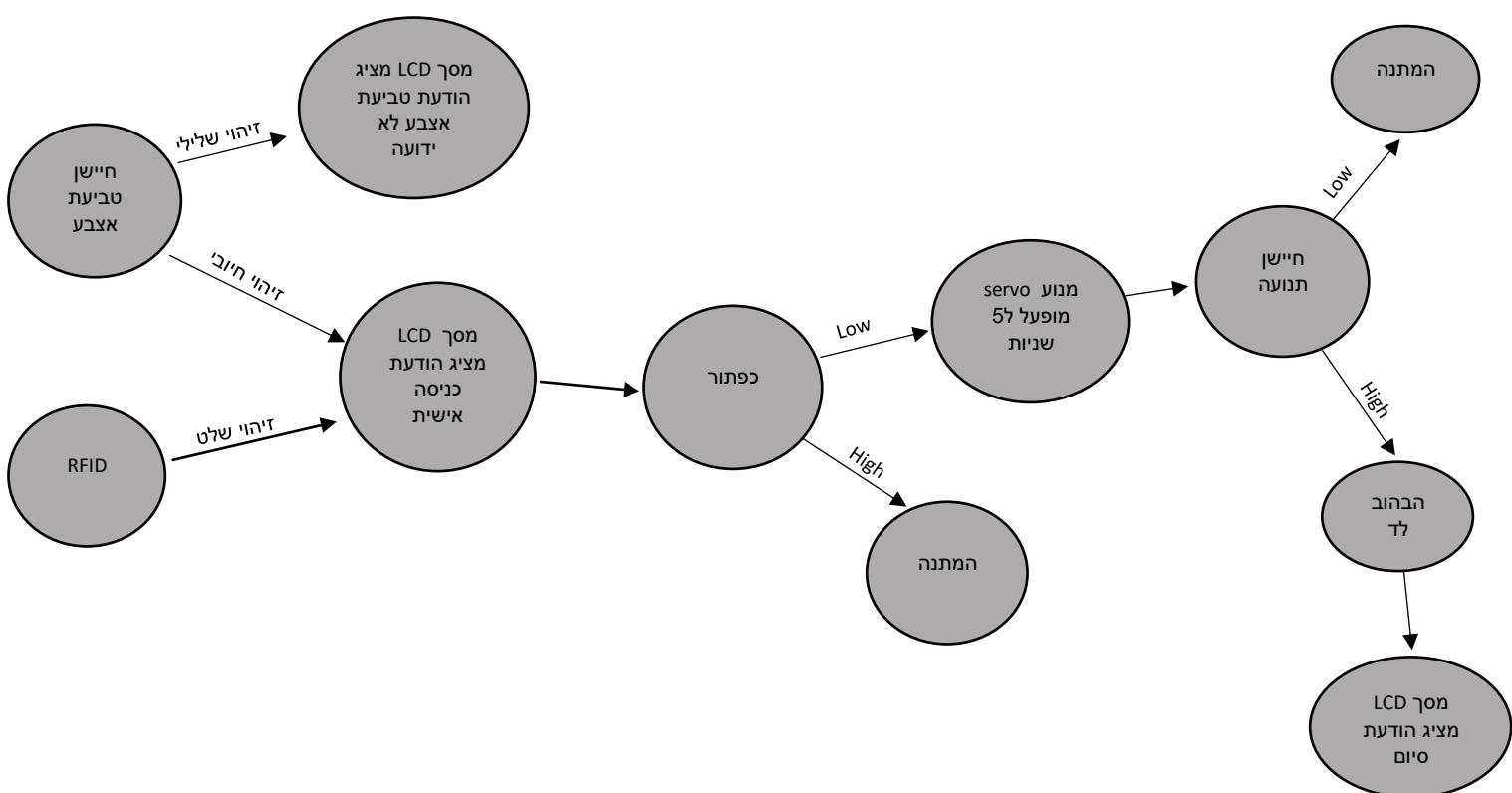
תיאור התוכן הכללי:

רכיב	מספר פין	תצורה	מטרת שימוש
חישון טביעה אצבע	4,5	input	זיהוי ביומטרי מאובטח של הבמאי לכניסת חדר הפעלה
חישון תקשורת RFID	Rst-9 SDA-10 MOSI-11 MISO-12 SCK-13 3.3v-3.3v	input	מנגנון גיבוי עוקף אבטחה במקרה של תקלת בחישון טביעת אצבע
LCD	A4,A5	output	הציג הודעת כניסה למפעיל
כפתור לחיצה	7	input	הפעלה ידנית של המנוע (הפעלת הבמה המסתובבת)
מנוע servo	3	output	ביצוע פעולה מכנית של סיבוב הבמה
חישון תנועה	2	input	זיהוי הגעת השחקן לנקודת סיום והפעלת חיוי סיום
נורת LED	6	output	חיוי ויזואלי על סיום המופע

תעודת זהות חיישן טביעה אצבע:

- שם הרכיב: AS608 Optical Fingerprint Sensor Module
- סוג חיישן: ביומטרי-אופטי (Optical)
- פרט טכני (Specifications):
- מתח עבודה: 5V DC3.3 – V.
- זרם עבודה: mA120 (זמן סריקה).
- marshak תקשורת: TTL Serial / UART (TTL Serial / UART).
- קצב העברת נתונים (Baud Rate): מוגדר כברירת מחדל ל-57600 bps (טווח 00-57600).
- זמן עיבוד תמונה: פחות מ-1.0 שניות.
- קיבולת זיכרון: אחסון של עד 162 תבניות טביעה אצבע.
- רמת דיק אבטחה:
- אחוז קבלת שגואה (FAR): קטן מ-0.001%.
- אחוז דחיה שגואה (FRR): קטן מ-1.0%.

תרשים זרימה:



בדיקות וגישות המערכת בתנאי סביבה שונים:

במסגרת הבדיקות, הרכנו את המערכת ב-5 תרחישים שונים כדי לבדוק את עמידות החישנים והלוגיקה.
להלן הממצאים:

1. **תאורת חדר רגילה:** המערכת תפקדה באופן אופטימלי. זיהוי טביעה האצבע והRFID היה מהיר, ומהסך היה קרייא.
2. **חסיפה לאור שימוש ישיר:** נצפתה רגישות גבוהה בחישון טביעת האצבע האופטי. האור הפריע לקריאה תקינה של האצבע בחלק מהניסיונות. כמו כן, תצוגת LCD הייתה קשה לקריאה ללא הצללה.
3. **סביבה חשוכה לחלווטין:** המערכת תפקדה מצוין. האור האחורי של LCD ושל חישון האצבע הספיקו לתפעול מלא. נורת LED יצירה אינדיקטיה ברורה מאוד.
4. **דיםות לחות:** נצפתה רגישות קרייטית. כאשר האצבע הייתה מעט לחה (זיעה או מים), חישון טביעת האצבע התקשה מאוד לבצע התאמה ודרש ניגוב של האצבע והחישון כדי להצליח.
5. **בדיקה אבטחה וזיהוי שגוי:** ניסינו לפתח את המערכת עם כרטיסים זרים (כמו רב-קו, כרטיס אשראי עם צ'יפ) ועם אצבעות של אנשים שלא נרשמו, המערכת הגיבה מצוין וחסמה את הגישה (הדף הודיעות שגיאה)

מתי זיהינו שהמערכת רגישה? מערכת הפגינה רגישות מיוחדת בשני מצבים עיקריים:

1. **תאורה קייזונית:** חישון טביעת האצבע האופטי רגיש מאוד לשינוי תאורה קייזוניים (שםן ישירה), שכן הוא מסתמן על צילום תמונה של האצבע.
2. **לחות וניקיון:** המערכת רגישה מאוד למצב הפיזי של האצבע הדוגמת. לכלה או רטיבות מנעו כניסה, מה שפוגע מעט בחווית המשתמש בתנאים לא סטריליים.

מסקנות מהעבודה:

הפרויקט המשיך כיצד מערכת בקרה פיזית תלולה באופן קרייטי בקשרו של המשתמש. נוכחנו לדעת שהמעבר בין השלבים חייב להיות מלאה בחווית ברור במסך LCD, אחרת המשתמש "הולך לאיבוד" בתחילת. בנוסף, השימוש של מספר אמצעי זיהוי (כרטיס וטביעת אצבע) הוכיח את עצמו כיעיל להנגשת המערכת למשתמשים שונים.

השדרוג שחברנו שכדי להוסיף למערכת הוא מגנון אimoto כפול. כרגע המערכת עובדת בתצורת "או/או" (כרטיס או אצבע). לשדרוג רמת האבטחה למתקנים נוספים, ניתן לשנות את הקוד כך שידרש את שניהם בראץ': המשתמש יצרך גם להעביר כרטיס וגם לסרוק אצבע כדי לקבל אישור כניסה, מה שיקשה מאוד על זיווף זהות.

נספח א': דף הוראות למפעיל

שם המערכת: בקרת כניסה (RFID + טביעת אצבע + חיישן תנועה)

1. מצב ההתחלתי

- כאשר המערכת דולקת וממתינה, נורת ה LED כביה.
- על צג ה LCD מופיע הכיתוב: SYSTEM LOCKED SCAN CARD/FINGER

2. שלב ההזיהוי והכניסה למשתמש יש שתי אפשרויות לביצוע ההזיהוי:

- **אפשרות א' - תג קירבה (RFID)**: יש להציג את התג הכחול/לבן לחישון הקורא.
- **אפשרות ב' - טביעת אצבע**: יש להניח את האצבע המוגדרת על משטח הסורק האופטי ולהמתין להבاهוב האדום.

- **במקרה של הצלחה**: המסר יציג "ROOM UNLOCKED" ולאחר מכן "WELCOME KEREN".
- **במקרה של כישלון**: המסר יציג "FINGER UNKNOWN" והמערכת תחזיר לנעילה.

3. פעולה בתוך החדר לאחר ההזיהוי המוצלח, המערכת ממתינה לפעולה יזומה:

- על המסר תופיע הודעה PRESS BUTTON
- יש ללחוץ על הלוחץ היעוד.
- תגובה המערכת : המנוע (Servo) יתחיל לפעול, ימתין 5 שניות וינעל חזרה.
- המסר יתעדכן לREADY FOR SENSOR

4. סיום תהליך יציאה כאשר המשתמש נע בחדר מול חיישן התנועה:

- המערכת תזהה תנועה ותפעיל רצף קבלת פנים: הבاهובי LED והציג שמות חברי הצוות על המסך.
- בסיום התצוגה, המערכת תבצע איפוס אוטומטי ותחזור במצב ההתחלתי (SYSTEM LOCKED) להמתינה למשתמש הבא.

נספח ב' : שימוש בכלי AI

במהלך פיתוח העבודה נעשה שימוש בכלי בינה מלאכותית לצורכי למידה ותמייה בלבד. נעזרנו בכלי ChatGPT ובכלי Gemini על מנת להבין את אופן החיבור הנוכחי בין רכיבי החומרה, לקבל סיוע בהבנת תקלות בקוד, ולהגדד את הלוגיקה של זרימת התוכנית. השימוש בכלים אלו נעשה כהכוונה כללית וכאמצע עזר, תוך כתיבה עצמאית של הקוד, קבלת החלטות טכניות וביצוע בדיקות בפועל על המערכת.

נספח ג' : תיאור קוד

הקוד נכתב בסביבת פיתוח IDE Arduino . המערכת עוברת בין מצבים שונים (נעול, פתוח, פעיל) בהתאם לקלט מהחישונים. להלן פירוט המבנה הלוגי והרכיבים:

1. ספירות והגדרות חומרה הקוד עושים שימוש בספריות חיצונית כדי לנוון את רכיבי החומרה השונים בפרוטוקולים מתאימים:

- הספרייה SPI.h או MFRC522.h משמשות לתקשורת עם רכיב ה-RFID בפרוטוקול SPI
- הספרייה LiquidCrystal_I2C.h או Wire.h משמשות לתקשורת עם מסך LCD בפרוטוקול I2C
- הספריות Adafruit_Fingerprint.h או SoftwareSerial.h משמשות ליצירת תקשורת טורית וירטואלית עם חיישן טביעה האצבע.
- הספרייה Servo.h משמשת לשיליטה במנוע הסרבו.

מיפוי החיבורים (Pinout) בלוח הארדואינו:

- חיישן RFID מחובר לפינים 9 (RST) ו-10 (SDA).
- חיישן טביעה אצבע מחובר לפינים 4 ו-5.
- מנוע הסרבו מחובר לפין 3.
- הקפטור מחובר לפין 7 מוגדר כקילט עם נגד משיכת פנימי.
- חיישן התנועה מחובר לפין 2.
- נורת ה-LED מחוברת לפין 6.

2. משתנים גלובליים לניהול מצבים הלוגיקה מנוהלת על ידי שני משתנים בוליאניים (daglis) המגדירים את מצב המערכת:

- המשתנה UnlockedIs מסמן האם המערכת עברה זיהוי מוצלח (אמת = המערכת פתוחה).
- המשתנה WasRunmotor מסמן האם בוצעה הפעולה הפיזית של פתיחת הדלת (כדי למנוע חזרה על הפעולה לפני סיום התהלהך המלא).

3. תיאור זרימת התוכנית (Main Loop Logic) הלולאה הראשית koop מחלוקת לשלווה שלבים מותנים:

שלב א: אימות כניסה חלק זה פועל רק כאשר המשתנה `isUnlocked` הוא שקר (המערכת נעולה). הקוד דוגם במקביל את חישון ה RFID ואת חישון טביעת האצבע. במקרה של זיהוי כרטיס תקין או טביעת אצבע תואמת, הקוד קורא לפונקציה `unlockSystem` שמשנה את מצב המשתנה `isUnlocked` לאמת.

שלב ב: הפעלה פיזית חלק זה פועל כאשר המערכת במצב פתוח `isUnlocked` לאמת (המנוע טרם הופעל). הקוד ממתין לסיגナル `LOW` מהכפטור (לחיצה). כאשר הכפטור נלחץ, מנוע הסרבו נעה לזרoit 0 מעלות, ממתין 5 שניות, וחוזר לזרoit 90 מעלות. לאחר מכן המשתנה `motorWasRun` הופך לאמת.

שלב ג: סיום ומשוב חלק זה פועל לאחר שהמנוע הופעל. הקוד ממתין לסיגナル `HIGH` מחישון התנועה. כאשר מזוהה תנועה, המערכת מבצעת סדרת הפעולות בנורית ה LED ומציגה את שמות המשתמשים על המסך. בסיום התצוגה, המערכת מפסיקת את כל המשתמשים הבוליאניים `isUnlocked` ו-`-motorWasRun` (חווזרים להיות שקר) וקורא לפונקציה שנעולת את המערכת.

4. פונקציות עזר מרכזיות:

- הפונקציה `getFingerprintID` אחראית על עיבוד התמונה מחישון האצבע והחזרת המזהה אם נמצא התאמה, או קוד שגיאה אם לא.
- הפונקציה `unlockSystem` מעדכנת את המסך להודעת פתיחה ומשנה את הסטטוס הלוגי של המערכת.
- הפונקציה `showLockedMessage` פונקציית אתחול המנקה את המסך ומציגה את הודעת הפתיחה הראשית המודיעיה שהמערכת נעולה.

נספח ד': קוד

```
#include <SPI.h>
#include <MFRC522.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <SoftwareSerial.h>
#include <Adafruit_Fingerprint.h>

// --- הגדרות פנימיות ---
#define RST_PIN 9
#define SS_PIN 10
#define BUTTON_PIN 7
#define SERVO_PIN 3
#define MOTION_PIN 2
#define LED_PIN 6

// פנים לטביעה אצבע
SoftwareSerial mySerial(4, 5);
```

```

// אובייקטים
MFRC522 mfrc522(SS_PIN, RST_PIN);
LiquidCrystal_I2C lcd(0x27, 16, 2);
Servo myServo;
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

// משתנים לוגיים
bool isUnlocked = false;
bool motorWasRun = false;

void setup() {
    Serial.begin(9600); // אתחול התקורתה ל-Serial Monitor
    SPI.begin();
    mfrc522.PCD_Init();
    finger.begin(57600);

    myServo.attach(SERVO_PIN);
    myServo.write(90);

    pinMode(BUTTON_PIN, INPUT_PULLUP);
    pinMode(MOTION_PIN, INPUT);
    pinMode(LED_PIN, OUTPUT);

    lcd.init();
    lcd.backlight();

    showLockedMessage();

    // הדפסה ראשונית: המערכת עלתה וモכנה
    Serial.println(F("System Initialized & Locked. Waiting for user input..."));
}

void loop() {
    // ======
    // או טביעת אצבע (RFID) המתנה לזרחי
    // ======
    if (!isUnlocked) {

        // בדיקת RFID
        if (mfrc522.PICC_IsNewCardPresent() && mfrc522.PICC_ReadCardSerial()) {
            // הדפסה: זהה כרטיס
            Serial.println(F("RFID Detected - Access Granted"));

            unlockSystem();
            mfrc522.PICC_HaltA();
            mfrc522.PCD_StopCrypto1();
        }

        // בדיקת טביעת אצבע
        int fingerResult = getFingerprintID();
    }
}

```

```

if (fingerResult >= 0) { // נמצאה התאמה
    // הדפסה: זהה אצבע תקינה מהה
    Serial.print(F("Fingerprint Match Found! ID: "));
    Serial.println(fingerResult);

    unlockSystem();
}
else if (fingerResult == -2) { // זהה אצבע לא מוכרת
    // הדפסה: אצבע לא מוכרת
    Serial.println(F("Access Denied - Unknown Fingerprint scanned"));

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("FINGER UNKNOWN");
    lcd.setCursor(0, 1);
    lcd.print("TRY AGAIN... ");
    delay(3000);
    showLockedMessage();
}
}

// =====
// שלב 2: מערכת פתוחה - כפטור ומנוע
// =====
if (isUnlocked && !motorWasRun) {
    if (digitalRead(BUTTON_PIN) == LOW) {
        // הדפסה: הקפטור נלחץ והמנוע מתחילה
        Serial.println(F("Button Pressed -> Motor Moving (0 degrees)..."));

        lcd.clear();
        lcd.print("Motor Running...");

        myServo.write(0);
        delay(5000);

        myServo.write(90);
        motorWasRun = true;

        // הדפסה: המנוע סימן
        Serial.println(F("Motor Finished. Waiting for Motion Sensor..."));

        lcd.clear();
        lcd.print("READY FOR SENSOR");
        delay(2000);
    }
}

// =====
// שלב 3: חישון תנועה - סיום עם קרדיטים

```

```

// =====
if (motorWasRun && digitalRead(MOTION_PIN) == HIGH) {
    // הדרישה: זיהה תנועה
    Serial.println(F("Motion Detected -> Starting Light Show & Credits sequence"));

    // 1. הבהובים הראשונים
    for (int i = 0; i < 5; i++) {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("SENSOR ACTIVATED");
        digitalWrite(LED_PIN, HIGH);
        delay(500);

        lcd.clear();
        lcd.setCursor(0, 1);
        lcd.print("LIGHTS ON LET'S GO");
        digitalWrite(LED_PIN, LOW);
        delay(500);
    }
    digitalWrite(LED_PIN, LOW); // וידוא כיובי

    // 2. הצגת השמות (חצי שנייה כל אחד)
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("ROMI"); delay(500);
    lcd.clear(); lcd.setCursor(0, 1); lcd.print("MOR"); delay(500);
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("HADAR"); delay(500);
    lcd.clear(); lcd.setCursor(0, 1); lcd.print("ALMOG"); delay(500);

    lcd.clear(); lcd.setCursor(0, 0); lcd.print("KEREN ?"); delay(500);
    lcd.clear(); lcd.setCursor(0, 1); lcd.print("KEREN !"); delay(500);
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("What?"); delay(500);
    lcd.clear(); lcd.setCursor(0, 1); lcd.print("KEREN !!"); delay(500);

    // 3. סיום הagi
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("AUTOMATION... ");
    delay(2000);

    lcd.setCursor(0, 1);
    lcd.print("BEST COURSE!");
    delay(5000);

    // 4. איפוס המערכת
    // הדרישה: איפוס המערכת להתחלה
    Serial.println(F("Sequence Complete. Resetting System..."));

    isUnlocked = false;
    motorWasRun = false;
    showLockedMessage();
}

```

```

}

// --- פונקציות עזר ---

void showLockedMessage() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("SYSTEM LOCKED");
    lcd.setCursor(0, 1);
    lcd.print("SCAN CARD/FINGER");
}

void unlockSystem() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("ROOM UNLOCKED");
    lcd.setCursor(0, 1);
    lcd.print("WELCOME KEREN"); // ניתן לשנות לשם המשתמש הרצוי

    isUnlocked = true;
    delay(2000);

    lcd.clear();
    lcd.print("PRESS BUTTON");
}

int getFingerprintID() {
    uint8_t p = finger.getImage();
    if (p != FINGERPRINT_OK) return -1;

    p = finger.image2Tz();
    if (p != FINGERPRINT_OK) return -1;

    p = finger.fingerFastSearch();
    if (p == FINGERPRINT_OK) {
        return finger.fingerID;
    } else {
        return -2;
    }
}

```