SIR SYED UNIVERSITY OF ENGINEERING & TECHNOLOGY
BIOMEDICAL ENGINEERING DEPARTMENT

MICRO PROCESSOR & INTERFACING PROJECT REPORT

Course Code: **BM-321L**
Semester: **5th**
Batch: **2022F**

Prepared By

Names	Roll Numbers
AMNA YOUSUF JUDGE	2022F-BBM-012
SHAHMEER HUSSAIN	2022F-BBM-014

Submitted To

Engr. S. M. Omair
Assistant Professor

Department: **Biomedical Engineering** Program: **B.S. (BM)**

Project
BM-321L Micro Processor & Interfacing

Announced date: 07-01-2025 Due Date: 22-01-2025 Total Marks = **15**
Marks Obtained =

Teacher Name**: S. M. Omair**

COMPLEX ENGINEERING ACTIVITY					
Lab CLO	**Lab Activity**	**Bloom's Taxonomy**	**PLO**	**Knowledge Profile**	**EA Attributes Addressed**
CLO 5	**Combines** resources to design various semester projects	A3 (Organization)	PLO 11 (WA11) Project Management	WK 5 Engineering Design	• Range of resources (EA1) • Level of interaction (EA2) • Familiarity (EA5)

DESIGN ACTIVITY:

Green-Line train owner requires a secure and automated VIP cabins for their customers. Design a password-based cabin access system having automated light & fan control, with duration of cabin occupation and temperature measurement.

Sir Syed University of Engineering & Technology, Karachi
Biomedical Engineering Department

Rubric for Semester Project Viva Evaluation

Distribution	Criteria	Exceeds Expectations (3.75)	Meets Expectations (3)	Developing (2.25)	Unsatisfactory (1.5)
Project Demonstration	CR1	Able to demonstrate the project with achievement of required objectives having clear understanding of project limitations and future enhancements. Hardware and/or Software modules are fully functional, if applicable.	Able to demonstrate the project with achievement of required objectives but understanding of project limitations and future enhancements is insufficient. Hardware and/or Software modules are functional, if applicable.	Able to demonstrate the project with achievement of at least 50% required objectives and insufficient understanding of project limitations and future enhancements. Hardware and/or Software modules are partially functional, if applicable.	Able to demonstrate the project with achievement of less than 50% required objectives and lacks in understanding of project limitations and future enhancements. Hardware and/or Software modules are not functional, if applicable.
Project results	CR2	Able to achieve all the desired results with alternate ways to improve measurements	Able to achieve all the desired results	Able to achieve most of the desired results with errors	Unable to achieve the desired results
Project Report	CR3	Project report has no grammatical and/ or spelling errors. All sections of the report are very well-written and technically accurate.	Project report has very few grammatical/ spelling errors. All sections of the report are technically accurate.	Project report has multiple grammatical/ spelling errors Few sections of the report contain technical errors.	Project report has several grammatical/ spelling errors and sentence construction is poor.
Viva	CR4	Able to answer the questions easily and correctly across the project.	Able to answer the questions related to the project	Able to answer the questions but with mistakes	Unable to answer the questions

Marks for Absentees = 0

Sir Syed University of Engineering & Technology, Karachi
Biomedical Engineering Department

Project Viva Evaluation

S. No.	Roll No.	Name	CR1	CR2	CR3	CR4	Total (15)
1	2022F-BM-012	AMNA YOUSUF JUDGE					
2	2022F-BM-014	SHAHMEER HUSSAIN					

Secure & Automated VIP Green-Line Cabins

Introduction:

The Secure & Automated VIP Green-Line Train Cabins project is designed to enhance the security, comfort, and efficiency of VIP cabins aboard Green-Line trains. The system employs an advanced password-based access control mechanism, where passengers must enter a correct password to gain access to the cabin. Upon entering the correct password, the system grants access, displaying an "Access Granted" message on the LCD, and simultaneously initiates the real-time display of the cabin's temperature and the length of stay timer.

Furthermore, the system integrates a Passive Infrared (PIR) motion sensor, which detects the presence or absence of occupants in the cabin. Upon detection of motion, the sensor triggers the activation of the cabin's lighting and fan, enhancing the passenger's comfort. The fan and lighting remain off when no motion is detected, ensuring energy efficiency. Importantly, the PIR sensor only becomes operational after access is granted, preventing any unauthorized control prior to entry.

This project focuses on providing a secure and user-friendly solution for managing cabin access and environmental conditions.

Aims:

- To create a secure system that restricts access to VIP cabins.
- To automate and manage cabin environment, including temperature detection, lighting and fans control, for a comfortable experience.
- To monitor and track cabin occupancy.
- To design a user-friendly interface for passengers and train staff.

Objectives:

- To design and implement a secure password authentication system to restrict access to VIP cabins, ensuring that only authorized individuals can enter.
- To integrate a temperature sensor that continuously monitors the cabin’s internal temperature, providing real-time data for effective environmental management.
- To control the cabin’s lighting and fan based on occupancy detection, utilizing a PIR sensor to activate or deactivate these devices based on the presence or absence of passengers.
- To track and display the length of stay in the cabin using a timer that updates in real-time, showing the elapsed time since access was granted.
- To develop a user-friendly interface on a 20x4 LCD screen, displaying critical information such as access status, temperature, and time, ensuring clarity and ease of use for passengers and train staff.

Methodology:

Materials:

Materials Used in the Secure & Automated VIP Green-Line Train Cabins Project:

1. Arduino Uno:
- Reason : The Arduino Uno serves as the main controller for managing sensors, the keypad, and the LCD. Its flexibility, ease of use, and extensive support make it ideal for this project.

2. 4x4 Keypad
- Reason : Used for password entry to secure access to the VIP cabin. It provides a simple and effective way for users to input their password.

3. 20x4 LCD Display
- Reason : Displays real-time system messages, cabin temperature, and timer information. The 20x4 format allows clear visibility of multiple lines of data at once.

4. Push Button
- Reason : Simulates the PIR sensor for detecting cabin occupancy. When pressed, it triggers the activation of the cabin's lighting and fan, representing the presence of a passenger.

5. LM35 Temperature Sensor
- Reason : Measures the cabin's temperature and sends the data to the Arduino, which displays it on the LCD. Its accuracy and ease of use make it suitable for this application.

6. Bar LEDs (for cabin lighting)
- Reason : Provides simulated lighting for the cabin. The LEDs are activated by the push button to simulate occupancy, enhancing the passenger experience.

7. DC Motor (used as fan)
- Reason : Simulates the fan in the cabin, controlled by the push button. It is used to provide additional comfort by turning on or off based on the simulated occupancy.

These materials were chosen for their compatibility, functionality, and ease of integration, ensuring a smooth and efficient automated system for the VIP cabin.

SYSTEM DESIGN:

Below is a detailed breakdown of the system's design and the sequence of operations:

1. System Initialization

Upon power-up, the system initializes, and the first task is to display the password prompt on the LCD screen. The LCD displays a message: "Enter Password:" to prompt the user to input their password.

2. Password-Based Access Control

1. Keypad Input: The system waits for the user to enter a password using a 4x4 keypad. Each key press is detected by the Arduino, which collects the input and stores it temporarily.

2. Password Validation: When the user presses the "Enter" or "=" key, the system compares the entered password with the pre-programmed correct password.

If the entered password is incorrect, the LCD displays a message like "Wrong Password". The user can then try again, and the system prompts for the password input once more.
If the entered password is correct, the LCD shows a confirmation message: "Access Granted". The system then proceeds to the next stage of operation.

3. Real-Time Data Display

Once access is granted, the following features are activated:

- Temperature Sensor Reading: The LM35 temperature sensor is continuously monitored. The sensor reads the current cabin temperature and converts the analog value to a digital temperature value. This temperature is displayed in real-time on the LCD in the format "Temp: XX.X C".
- Length of Stay Timer: Simultaneously, the system starts tracking the length of stay by counting the time since access was granted. The timer begins counting seconds from the moment of access, and the current elapsed time is displayed on the LCD in the format "Time: HH:MM:SS".
- The timer updates every second. Once 60 seconds elapse, the system automatically increments the minutes, and once 60 minutes pass, it increments the hours. This ensures that the time is displayed accurately, including rollover for minutes and hours.

4. Occupancy Detection and Control (Push Button as PIR Simulation)

After access is granted, the system is ready to simulate occupancy detection:

- Push Button Simulation: A push button is used to simulate the detection of passenger occupancy. When the button is pressed (i.e., when the input is HIGH), it mimics the activation of a PIR motion sensor indicating the presence of a passenger in the cabin.
- When the Push Button is Pressed (Simulating Occupancy): The system turns on both the fan (simulated by a DC motor) and the cabin lighting (controlled via bar LEDs). This action is meant to simulate the environment when a passenger is detected inside the cabin.
- When the Push Button is Released (Simulating Absence of Occupancy): The system turns off both the fan and the bar LEDs. This action mimics the environment when no passenger is detected, keeping the cabin energy-efficient when unoccupied.

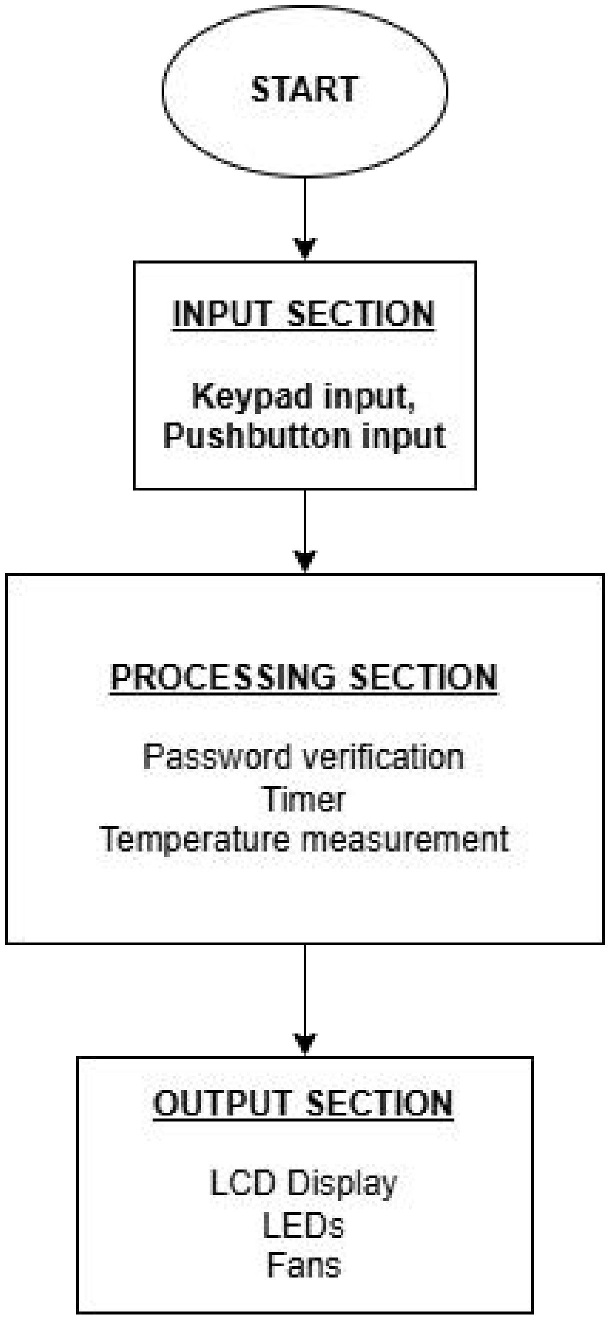
5. User Feedback

The system provides feedback to the user on the LCD during each of these stages:

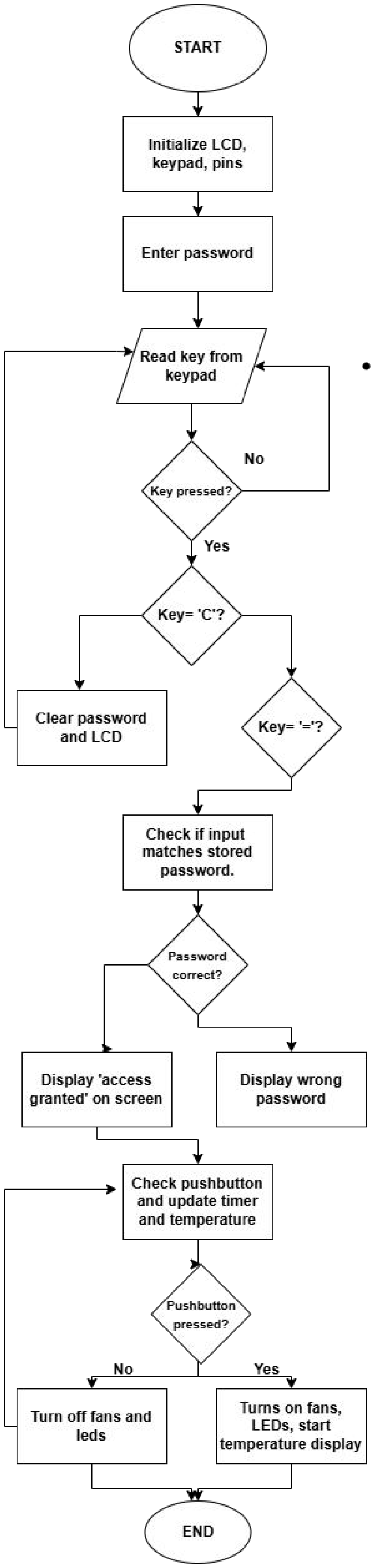
- If an incorrect password is entered, the user is immediately informed with a "Wrong Password" message on the LCD.
- Upon successful password entry, the LCD displays "Access Granted", followed by the real-time temperature and timer.

The system also continuously updates the time and temperature on the LCD as long as access is granted. The status of the fan and lighting is displayed based on the push button press. When the button is pressed, the fan and lights are activated, and this is reflected in the cabin environment

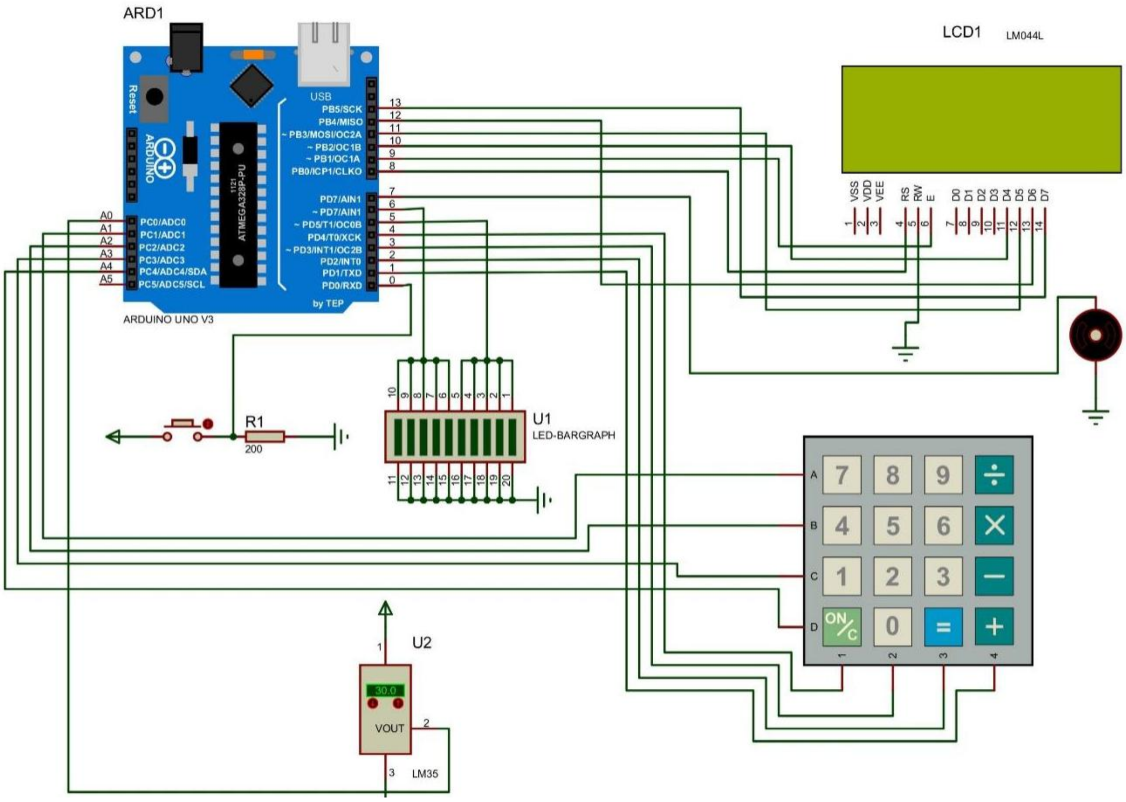
System block diagram:



Program flowchart.

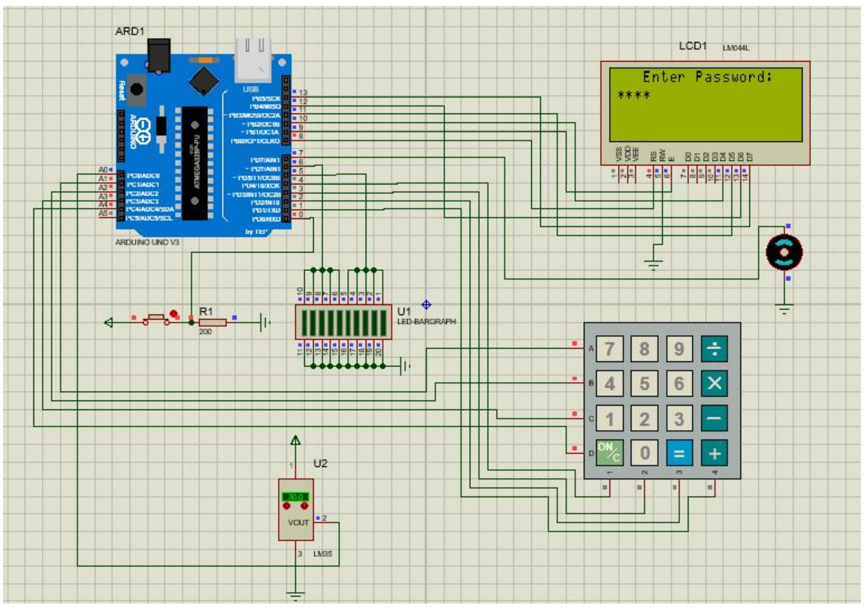


Circuit diagram (Proteus Schematic):



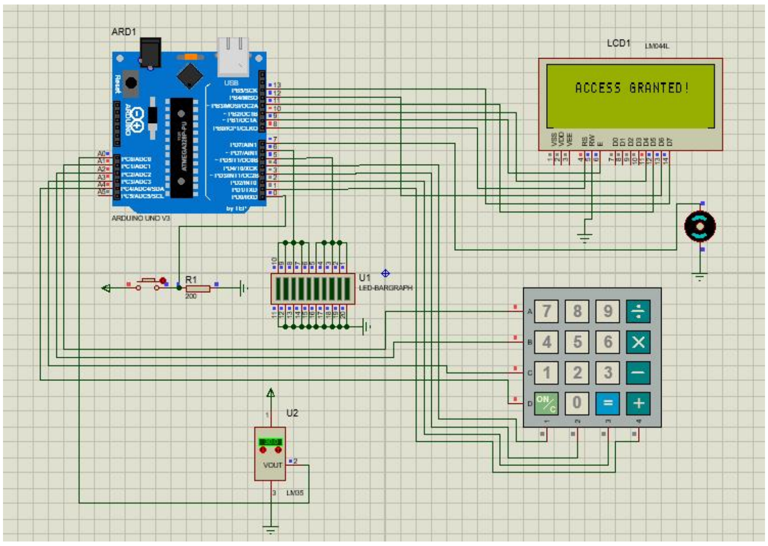
STEPS: (Proteus Simulations)

START

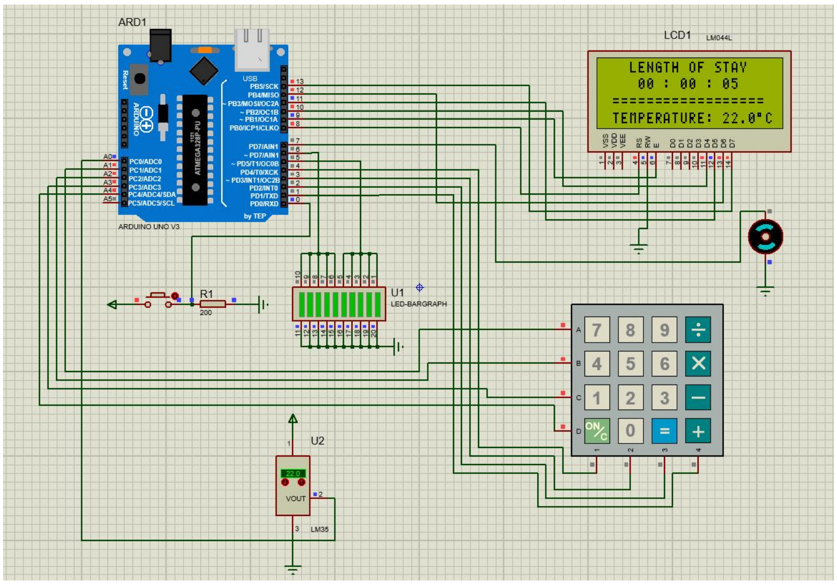


RESULTS:

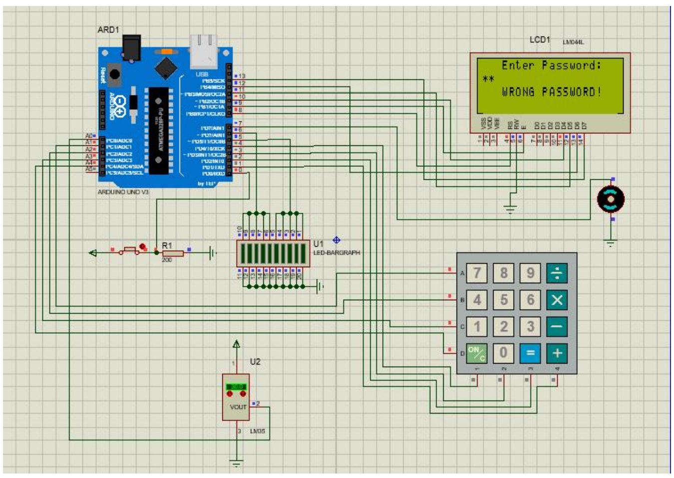
ACCESS GRANTED (CORRECT PASSWORD):



BUTTON PRESSED:



ACCESS DENIED (WRONG PASSWORD)



Source Code (ARDUINO):

```
#include <LiquidCrystal.h>
#include <Keypad.h>
LiquidCrystal lcd(8, 9, 10, 11, 12, 13);  // LCD pin connections
const byte pushbutton = 0;
const byte ledPin = 5;
const byte ledPin2 = 6;
const int fanPin = 7;
const int tempPin = A0;
const String password = "1214";  // Change as needed
String inputPassword = "";
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] =
{ {'7', '8', '9', '/'},
  {'4', '5', '6', '*'},
  {'1', '2', '3', '-'},
  {'C', '0', '=', '+'}  };
byte rowPins[ROWS] = {A1, A2, A3, A4};
byte colPins[COLS] = {4, 3, 2, 1};
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);
void setup()
{ lcd.begin(20, 4);
  lcd.setCursor(3, 0);
  lcd.print("Enter Password:");
  pinMode(fanPin, OUTPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(tempPin, INPUT);
  pinMode(pushbutton, INPUT);
  digitalWrite(fanPin, LOW);  // Fan OFF initially
  digitalWrite(ledPin, LOW);  // LED OFF initially
  digitalWrite(ledPin2, LOW); // LED OFF initially
  }
void loop()
{  char key = keypad.getKey();
  if (key) {
    if (key == 'C') {  // Clear input
      inputPassword = "";  // Reset password
      lcd.setCursor(0, 1);
      lcd.print("                    "); }
    else if (key == '=') {
      if (inputPassword == password) {
        lcd.setCursor(0, 0);
        lcd.print("                    ");  // Clear enter password
        lcd.setCursor(0, 1);
        lcd.print("                    ");  // Clear password
        lcd.setCursor(3, 1);
        lcd.print("ACCESS GRANTED!");
        delay(800);
        lcd.setCursor(3, 1);
```

```
    lcd.print("                "); // Clear Access Granted message
    unsigned long startTime = millis(); // Start the timer
    unsigned long lastUpdate = millis();
    int hours = 0, minutes = 0, seconds = 0;
    while (true) {
      unsigned long currentTime = millis();
      if (currentTime - lastUpdate >= 1000) {
        lastUpdate += 1000;
        seconds++;
        if (seconds == 60) {
          seconds = 0;
          minutes++;   }
        if (minutes == 60) {
          minutes = 0;
          hours++; }
        lcd.setCursor(3, 0);
        lcd.print("LENGTH OF STAY");
        lcd.setCursor(4, 1);
        lcd.print(hours < 10 ? "0" : ""); // Add leading zero
        lcd.print(hours);
        lcd.print(" : ");
        lcd.print(minutes < 10 ? "0" : ""); // Add leading zero
        lcd.print(minutes);
        lcd.print(" : ");
        lcd.print(seconds < 10 ? "0" : ""); // Add leading zero
        lcd.print(seconds);
        lcd.setCursor(1, 2);
        lcd.print("=================="); // Display divider }
      int tempValue = analogRead(tempPin);
      int check = digitalRead(pushbutton);
      if (check == 1) {
        digitalWrite(fanPin, HIGH);
        digitalWrite(ledPin, HIGH);
        digitalWrite(ledPin2, HIGH);
        float temperature = (tempValue * 5.0 / 1023.0) * 100;
        lcd.setCursor(1, 3);
        lcd.print("TEMPERATURE: ");
        lcd.print(temperature, 1); // Display with 1 decimal place
        lcd.write(223); // ascii for degree symbol
        lcd.print("C"); }
      else {
        digitalWrite(fanPin, LOW);
        digitalWrite(ledPin, LOW);
        digitalWrite(ledPin2, LOW);   }}
        }
}         else {
    lcd.setCursor(3, 2);
    lcd.print("WRONG PASSWORD!");
    delay(2000);   }
  inputPassword = "";
  lcd.setCursor(3, 2);
  lcd.print("                "); }
```

```
else {
  if (inputPassword.length() < 4) {
    inputPassword += key;
    lcd.setCursor(0, 1);
    String maskedPassword = "";
    for (int i = 0; i < inputPassword.length(); i++) {
      maskedPassword += "*";   }
    lcd.print(maskedPassword + " ");    }}}}
```

Conclusion:

We designed a helpful and user-friendly system for passengers, offering features such as password-protected access, temperature monitoring, and length-of-stay tracking. Its simple and efficient design makes sure that passengers can comfortably and conveniently use the system, which in turn makes their overall experience much better. In creating this project, we applied various techniques and skills learned in our Microprocessor Interfacing labs to design both the code and the circuit. These skills enabled us to integrate hardware components like the LCD, keypad, temperature sensor, and LEDs with the Arduino UNO as well as simulating a push button as a motion sensor.

This project also was an opportunity to better our understanding of flowchart and block diagram creation. By employing methods from different sources, we were able to design a clear and functional system representation, which greatly helps in understanding the process. Additionally, we gained valuable experience in interfacing with an Arduino, when faced with the challenge of using minimal pins. Since only one Arduino was employed in Proteus, we had to carefully assign and optimize each pin while ensuring that all functions were properly interfaced. This process required us to thoroughly understand the purpose and functionality of each Arduino pin and how to use them effectively.

Overall, this project helped us enhance our practical knowledge of Proteus simulation and Arduino interfacing. It provided a comprehensive learning experience by combining theory with practical implementation, helping increase our problem solving and technical experience. The result is a clean and functional system that shows our ability to design and implement such integrated circuits for real world applications.

Limitations:

· **Lack of Backup Power Supply**: The system does not include any backup power, such as a battery, which means it cannot function during a power outage.

· **Limited Temperature Monitoring**: The temperature monitoring feature relies on basic sensors with limited accuracy and does not include any advanced features like alarms for extreme temperature conditions, it can be solved by using DHT11.

· **Single-Cabin Focus**: The project is designed for a single cabin and lacks scalability to accommodate multiple cabins or a networked system.

· **No Remote Access or Control**: The system lacks connectivity features such as Wi-Fi or Bluetooth for remote monitoring and control.

· **No Emergency Override Mechanism**: The project does not include an emergency override feature for situations where access needs to be granted immediately.