National Textile University

Department of Computer Science

Name:	Fareena Shahbaz Fatima Shahzad Fiza Mahboob
Class:	BSCS -A (5th Semester)
Registration No:	23-NTU-CS-1024 23-NTU-CS-1026 23-NTU-CS-1027
Course Code:	COC-2073
Course Name:	Embedded IOT system
Submitted To:	Sir Nasir Mehmood
Submission Date:	January 2nd ,2026

Project Title:
SMART STADIUM:

Automated Crowd & Environment Control

Contents

What is Smart Stadium?

In modern stadiums, managing large crowds efficiently is critical to ensure safety, save energy, and provide a comfortable experience. Traditional systems rely on manual monitoring for counting people, controlling lighting, and checking environmental conditions. This approach is error-prone, inefficient, and unable to provide real-time updates.

The **Smart Stadium System** is an **IoT-based solution** designed to automate crowd management, lighting, and environmental monitoring. By integrating sensors with the ESP32 microcontroller, the system counts visitors, controls lights automatically, monitors temperature and humidity, and sends real-time data to a mobile application using the **Blynk platform**. This reduces manual labor, improves safety, and enhances energy efficiency.

Problem Statement

Conventional stadium management systems face several challenges:

1. **Manual Person Counting:** Human counting is inaccurate and inefficient, especially in large venues.
2. **Manual Lighting Control:** Lights are often left on unnecessarily, wasting electricity.
3. **No Real-Time Monitoring:** Managers cannot monitor crowd size or environmental conditions remotely.
4. **Safety Risks:** Without automated entry control, overcrowding can occur, leading to potential hazards.

Objectives

The **Smart Stadium System** aims to:

- Automatically count visitors entering the stadium.
- Control entry gates when the maximum crowd limit is reached.
- Automate lighting based on surrounding light intensity.
- Monitor temperature and humidity to ensure a comfortable environment.
- Send real-time data to a mobile application for remote monitoring.

This project enhances safety, reduces energy consumption, and provides an intelligent user-friendly experience.

Proposed Solution

The system uses the following approach:

- **ESP32 Microcontroller:** Acts as the central control unit with built-in Wi-Fi for IoT communication.
- **IR Sensor:** Detects and counts individuals entering the stadium.
- **LDR Sensor:** Measures ambient light and automatically turns LEDs ON or OFF.
- **DHT11 Sensor:** Monitors temperature and humidity.
- **Servo Motor:** Controls the entry gate automatically.
- **Buzzer:** Alerts when someone enters and when the stadium reaches maximum capacity.
- **OLED Display:** Shows real-time system information.
- **Blynk App:** Enables remote monitoring through a mobile phone.

Hardware Components :

1. ESP32 Microcontroller

- Features: Built-in Wi-Fi, multiple GPIO pins, suitable for IoT applications.

2. Sensors

- **IR Sensor:** Counts people entering.
- **LDR Sensor:** Detects light intensity to control LEDs.
- **DHT11 Sensor:** Measures temperature and humidity.

3. Actuators

- **Servo Motor:** Controls the gate.
- **Buzzer:** Alerts for person entry and full capacity.
- **LEDs:** Automatic lighting.

4. Breadboard

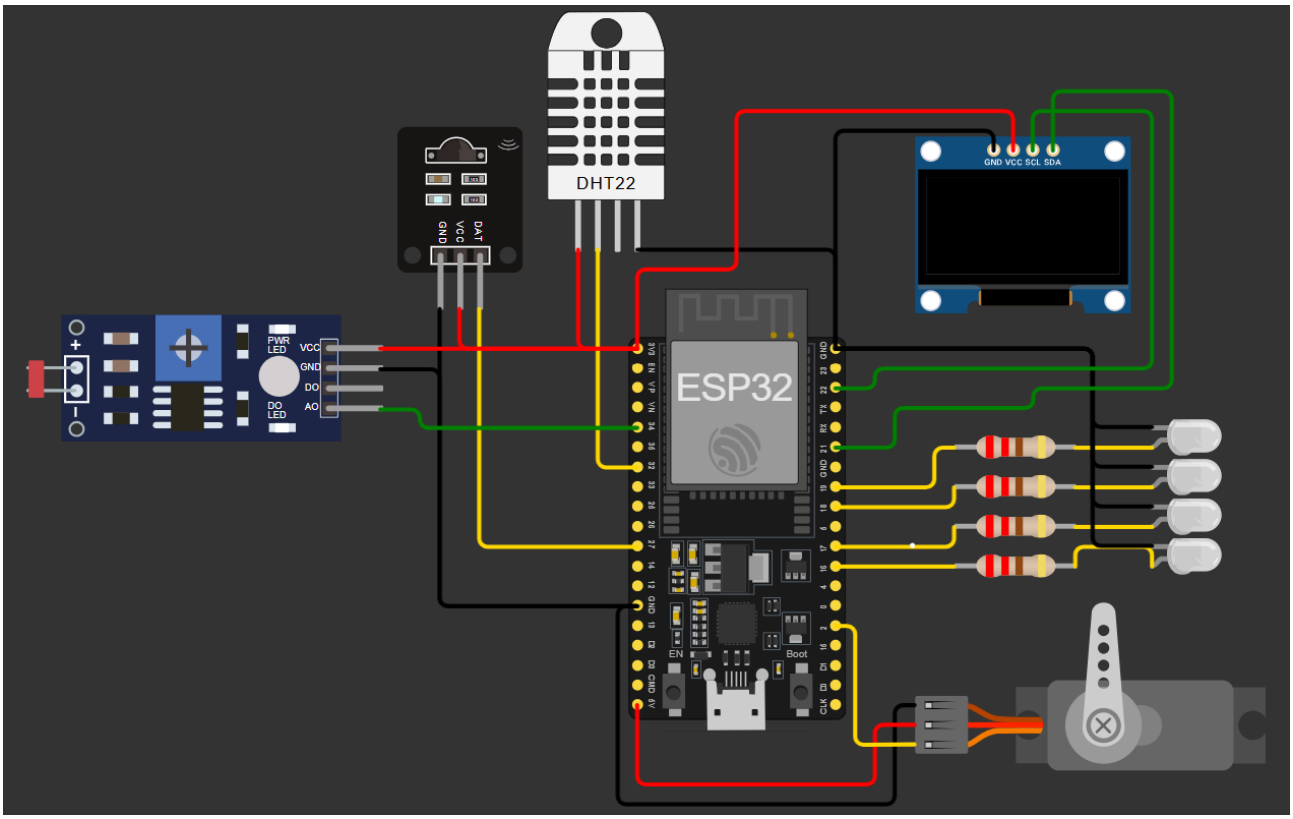
Software Technologies used:

- **Visual Studio Code (VS Code)**
    - Code editor for writing, editing, and managing ESP32 programs efficiently.
- **Wokwi Simulator**
    - Simulates and tests ESP32 circuits online before actual hardware implementation.
- **Blynk IoT Platform**
    - Provides remote monitoring and visualization of live data on a mobile device.

Pin Configuration

Component	**ESP32 Pin**	**Purpose**
IR Sensor	GPIO 27	Person detection
LDR	GPIO 34	Light sensing
DHT11	GPIO 32	Temperature & humidity sensor
Servo Motor	GPIO 2	Gate control
Buzzer	GPIO 26	Alert sound
LEDs	GPIO 16 GPIO 17 GPIO 18 GPIO 19	Automatic lighting
OLED SDA	GPIO 21	I2C data
OLED SCL	GPIO 22	I2C clock

Block Diagram



Basic features:

1. Automatic Person Counting

- IR sensor detects a person crossing the gate.
- Counter increments by 1 for each detection.

2. Crowd Limit Control

- Maximum capacity is set (e.g., 20 persons).
- Servo motor closes the gate when limit is reached.

3. Automatic Lighting System

- LDR reads ambient light levels.
- Lights turn ON in darkness and OFF in bright conditions.

4. Temperature and Humidity Monitoring

- DHT11 sensor measures environment.
- Data displayed on OLED and Blynk app.

5. IoT Monitoring using Blynk

- ESP32 sends sensor data to Blynk app every few seconds.
- Users can monitor real-time data remotely.

6. OLED Display

- Shows person count, temperature, humidity, and system status.

Methodology

The Smart Stadium system is implemented using an embedded IoT-based approach. The working methodology is divided into the following steps:

1. **System Initialization**

The ESP32 initializes all connected sensors, actuators, OLED display, Wi-Fi, and the Blynk cloud platform.

2. **Crowd Detection**

An IR sensor detects each person entering the stadium. The ESP32 increments the count and checks it against the predefined maximum limit.

3. **Gate Control & Alerts**

When the crowd limit is reached, the servo motor closes the gate and the buzzer generates an alert to indicate full capacity.

4. **Automatic Lighting**

The LDR sensor measures ambient light intensity. LEDs are automatically turned ON in low light and OFF in bright conditions.

5. **Environmental Monitoring**

The DHT11 sensor continuously monitors temperature and humidity for environmental comfort.

6. **Data Display & Cloud Monitoring**

All real-time data is displayed on the OLED screen and simultaneously sent to the Blynk cloud for remote monitoring via a mobile application.

System Workflow

1. **Initialization:**
    - ESP32 initializes sensors, OLED, WiFi, and Blynk.

2. **Loop Operation:**
    - Continuously read IR, LDR, and DHT11 sensors.
    - Update person count and environmental readings.
    - Control gate and LEDs automatically.
    - Send real-time data to the Blynk app.
3. **Alerts:**
    - Short beep for each person entry.
    - Long beep and gate closure when maximum capacity is reached.

Programming language:

We have used programming language **C++.**

Libraries:

- <Arduino.h> – Core Arduino functions.
- <WiFi.h> – WiFi connectivity for ESP32.
- <WiFiClient.h> – Client-server communication.
- <BlynkSimpleEsp32.h> – IoT communication with Blynk app.
- <Wire.h> – I2C communication.
- <Adafruit_GFX.h> – Graphics library for OLED.
- <Adafruit_SSD1306.h> – Controls OLED display.
- <DHT.h> – DHT11 temperature & humidity sensor.
- <ESP32Servo.h> – Servo motor control.

Major Code Functions

- longBeep() – Produces a long alert when stadium reaches full capacity.
- shortBeep() – Short beep when a person enters.
- updateOLED(temp, hum) – Updates the OLED display with current data.
- checkLighting() – Controls LEDs based on LDR readings.

- sendToBlynk(temp, hum) – Sends sensor data to the Blynk app.

Code Workflow

1. **Header Files and Libraries**

   Include required libraries for sensors, display, WiFi, and IoT.

2. **Pin Configuration**

   Assign GPIO pins for IR sensor, LDR, buzzer, servo motor, LEDs, and DHT11.

3. **Setup Function**

   Initializes sensors, OLED display, WiFi connection, and Blynk platform.

4. **Loop Function**

   Continuously checks sensors, updates display, controls lighting, counts people, and sends data to Blynk

Future Scope

- Integration of camera-based AI crowd counting
- Advanced cloud data analytics and reports
- Mobile alert notifications for emergencies
- Expansion to multiple entry gates
- Use of renewable energy sources like solar panels

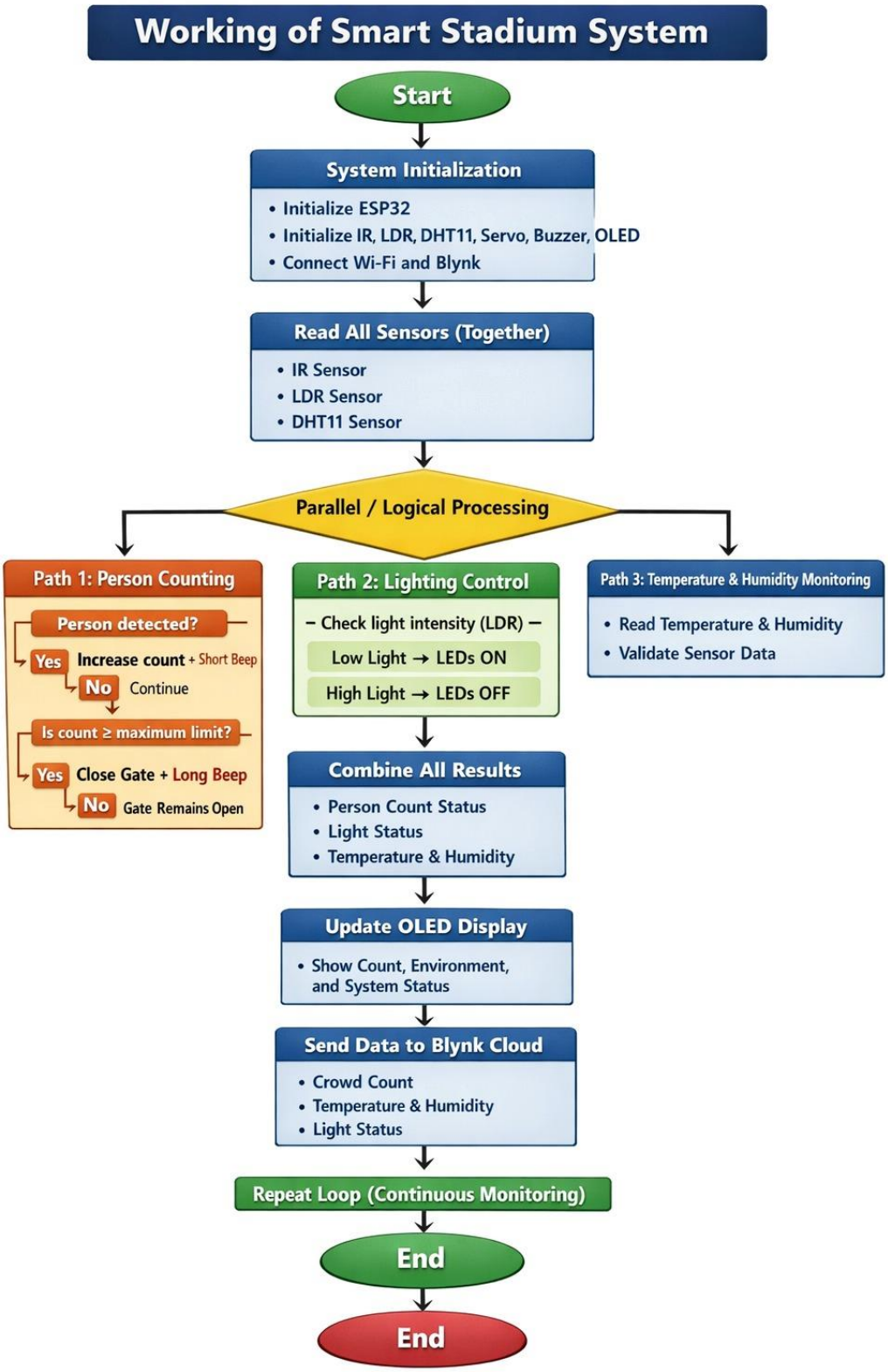
Advantages

- **Automation:** Reduces human effort for counting and monitoring.
- **Safety:** Prevents overcrowding through automatic gate control.
- **Energy Efficiency:** Lights turn on/off automatically based on environment.
- **Real-Time Monitoring:** Provides live updates through a mobile application.
- **Scalable:** Can be expanded to larger stadiums with multiple gates.

Applications

- Stadiums and Sports Arenas
- Auditoriums and Concert Halls
- Event Venues and Malls
- Smart Buildings

Flowchart


Working of Smart Stadium System
Start
System Initialization
• Initialize ESP32
• Initialize IR, LDR, DHT11, Servo, Buzzer, OLED
• Connect Wi-Fi and Blynk
Read All Sensors (Together)
• IR Sensor
• LDR Sensor
• DHT11 Sensor
Parallel / Logical Processing
Path 1: Person Counting
Person detected?
Yes
Increase count + Short Beep
No
Continue
Is count ≥ maximum limit?
Yes
Close Gate + Long Beep
No
Gate Remains Open
Path 2: Lighting Control
– Check light intensity (LDR) –
Low Light → LEDs ON
High Light → LEDs OFF
Path 3: Temperature & Humidity Monitoring
• Read Temperature & Humidity
• Validate Sensor Data
Combine All Results
• Person Count Status
• Light Status
• Temperature & Humidity
Update OLED Display
• Show Count, Environment, and System Status
Send Data to Blynk Cloud
• Crowd Count
• Temperature & Humidity
• Light Status
Repeat Loop (Continuous Monitoring)
End
End

Results:

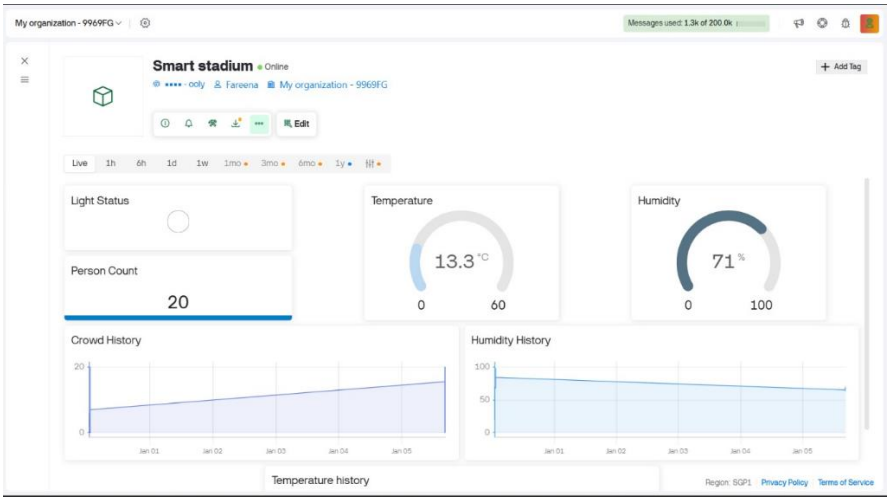
- Accurate automatic person counting using IR sensor
- Successful prevention of overcrowding by gate control
- Automatic lighting reduced unnecessary energy usage
- Real-time temperature and humidity monitoring achieved
- Live data monitoring through Blynk cloud application

Conclusion

In The **Smart Stadium System** demonstrates an effective use of IoT and embedded systems in real-world applications. By integrating **ESP32, sensors, actuators, and IoT platforms**, the system achieves automated crowd control, lighting management, and environmental monitoring. It enhances safety, saves energy, and allows stadium managers to monitor operations remotely in real-time.

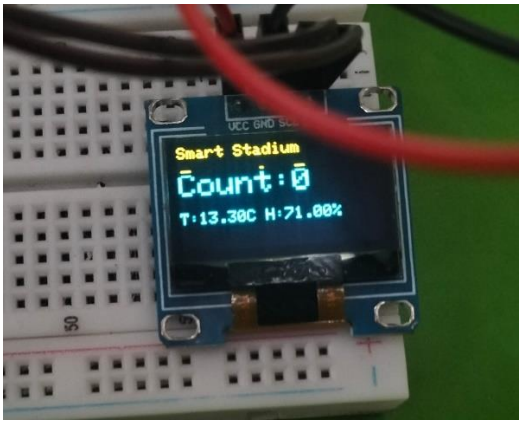
This project serves as a practical implementation of embedded system concepts, demonstrating how IoT can revolutionize traditional infrastructure.

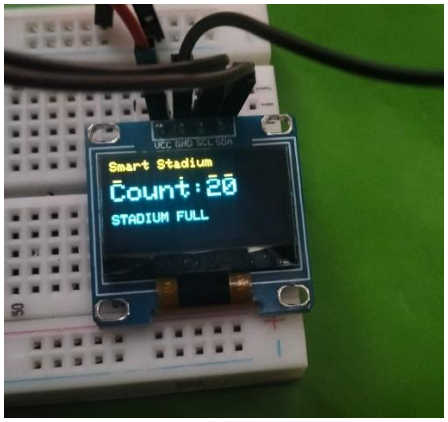
Blynk Screenshots



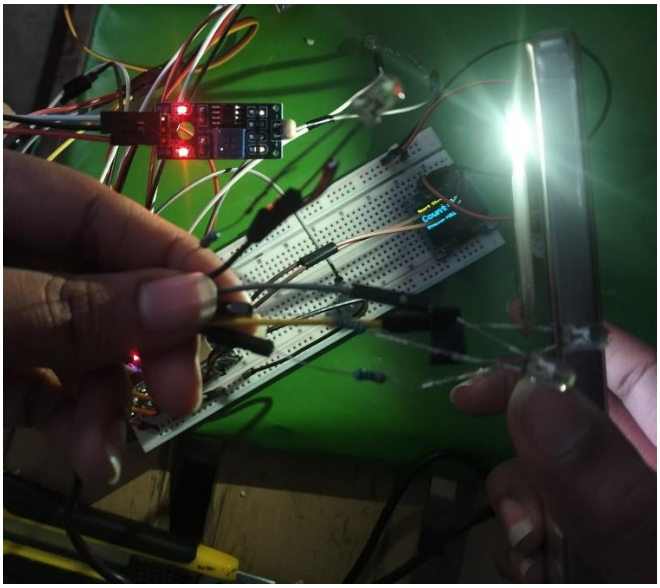
Working pictures

Counting feature:

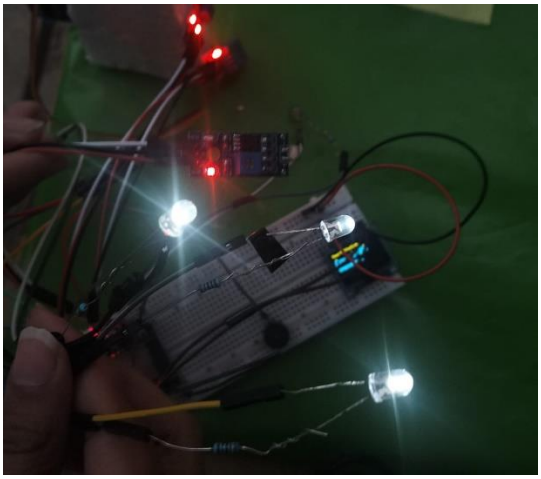




In high light the lights turn off:



And in low light the lights turn on:



The gate closes when the count reaches the maximum capacity: