A REPORT OF FOUR WEEK TRAINING (14 PT.)

at

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD

OF THE DEGREE OF (12pt.)

BACHELOR OF TECHNOLOGY (14 pt.)

(Computer Science and Engineering) (14pt.)

JUNE-JULY, 2025 (14 pt.)

SUBMITTED BY:

Kashish Gujral-2302584

Kareena-2302582

Kajal-2302575

Lovejot Kaur-2302596

(12pt.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (12pt.)

GURU NANAK DEV ENGINEERING COLLEGE LUDHIANA (14pt.)

CERTIFICATE BY INSTITUTE

GURU NANAK DEV ENGINEERING COLLEGE

An Autonomous College under UGC Act 1956

GILL PARK, GILL ROAD, LUDHIANA - 141006 (PUNJAB) INDIA

AICTE Approved, ISO: 9001-2015 Certified, Affiliated to IK Gujral, PTU, Kapurthala
IEI Accredited, UG Programes, Institute Accredited by NAAC ('A' Grade) & TCS

TESTING AND CONSULTANCY CELL

Website : http://tcc.gndec.ac.in E-mail : tcc@gndec.ac.in

This is certified that Ms./Mr. Kashish Gujral *D/o, S/o* Arvind Kumar Gujral

Roll No. 2302584 *of college* Guru Nanak Dev Engineering College

branch Civil Engg. / Mechanical Engg. / Electronics & Communication Engg. / Electrical Engg. / Computer Science & Engg. / Information Technology has successfully completed participated Industrial Training / Institutional Training Programme on SMART FUSION (AI-ML-IoT)

from 23rd June, 2025 *to* 21st July, 2025

Co-ordinator Head of the Department Dean (T & C) Principal

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

CANDIDATE'S DECLARATION

I KASHISH GUJRAL hereby declare that I have undertaken four week training GURU NANAK DEV ENGINEERING COLLEGE during a period from __25-6-25__ to ___18-7-25____ in partial fulfillment of requirements for the award of degree of B.Tech. (Computer Science and Engineering) at Guru Nanak Dev Engineering College, Ludhiana. The work which is being presented in the training report submitted to Department of Computer Science and Engineering at Guru Nanak Dev Engineering College, Ludhiana is an authentic record of training work.

Name Of Students	Signature of Students
1. Kashish Gujral	
2. Kareena	
3. Kajal	
4. Lovejot Kaur	

The four week industrial training Viva–Voce Examination of_ SMARTFUSION(AI-ML-IOT)__ has been held on ____16-10-25________ and accepted.

Signature of Internal Examiner   Signature of External Examiner

ABSTRACT

This report summarizes the work carried out during the four-week industrial training at Smart Fusion, Guru Nanak Dev Engineering College, Ludhiana. The primary objective of our project was to design and implement "AgroSolAR – Automated Rotation", a sunflower solar tracking system using the ESP32 microcontroller, Light Dependent Resistors (LDRs), and IoT technology.

The system is designed to maximize solar energy collection by automatically aligning the solar panel towards the direction of maximum sunlight. During the training, we developed the hardware and wrote the firmware to interface ESP32 with sensors and motors. Real-time monitoring and control were achieved through IoT integration.

Our prototype showed a significant improvement in energy harvesting efficiency compared to fixed solar panels. This project demonstrates the practical application of AI, ML, and IoT in the field of renewable energy. The experience gained helped us understand embedded systems design and teamwork in an industrial environment.

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to **Smart Fusion** and **Guru Nanak Dev Engineering College, Ludhiana,** for providing us with the opportunity to undertake the four-week training and work on **the “AgroSolAR – Automated Rotation”** project.

We are thankful to our professor, training coordinator, and all faculty members for their valuable guidance, encouragement, and support during the project. We also extend our thanks to our group members and friends for their cooperation and teamwork.

The experience added to our practical knowledge in computer science and engineering, and we are grateful for this learning opportunity.

ABOUT SMART FUSION (TRAINING PROGRAM)

Smart Fusion is a technology-driven training institute that offers practical skill development in the fields of computer science, AI, IoT, and machine learning. The four-week training program for B.Tech students at Guru Nanak Dev Engineering College focuses on industrial applications, hands-on projects, and exposure to current technological trends.

During our training, we learned about microcontroller development, sensor interfacing, and AI/ML implementation through our group project—AgroSolAR (Automated Rotation). The program's emphasis on teamwork, project-based learning, and industry interaction helped us gain practical knowledge to support our academic studies.

LIST OF FIGURES

CONTENT

CHAPTER - 1

INTRODUCTION

1.1 BACKGROUND OF THE TRAINING

Industrial training bridges the gap between academic concepts and real engineering practice, equipping students with the skills demanded in industry today. Guru Nanak Dev Engineering College, Ludhiana's four-week "SMARTFUSION" training (specializing in AI, ML, and IoT) was designed to provide both theoretical grounding and hands-on experience in modern smart technologies.

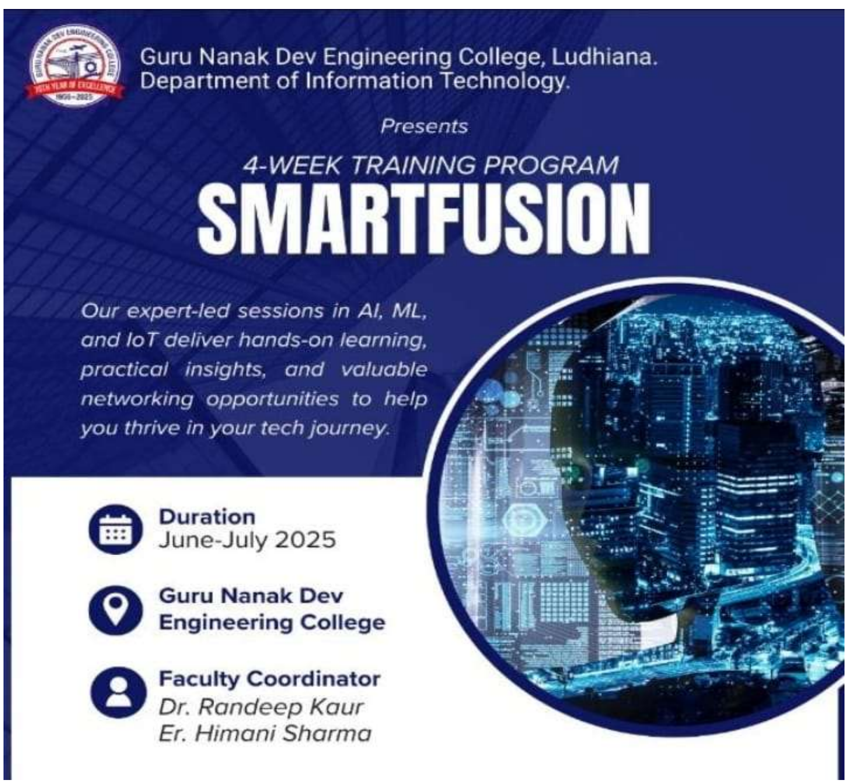


Figure 1.1: Official Poster of SMARTFUSION Training Program

Industrial advancement relies on the integration of intelligent automation and data-driven applications. AI, ML, and IoT are the pillars enabling such advancements—making it crucial for engineering graduates to master these domains.

1.2 THEORETICAL EXPLANATION OF KEY TECHNOLOGIES

1.2.1 ARTIFICIAL INTELLIGENCE (AI)

AI refers to technologies that allow machines to perform tasks requiring human-like intelligence—reasoning, problem-solving, learning, and adapting. Popular AI domains include natural language processing, robotics, and expert systems.

1.2.2 MACHINE LEARNING (ML)

ML is a subset of AI focused on designing and applying algorithms that learn from and make predictions based on data. Supervised, unsupervised, and reinforcement learning represent major categories—each having broad applications in areas such as image classification, speech recognition, and predictive analytics.

1.2.3 INTERNET OF THINGS (IOT)

IoT involves connecting physical devices (sensors, appliances) to the internet, enabling them to collect and share data. IoT systems power smart cities, industrial monitoring, automation, and home convenience.

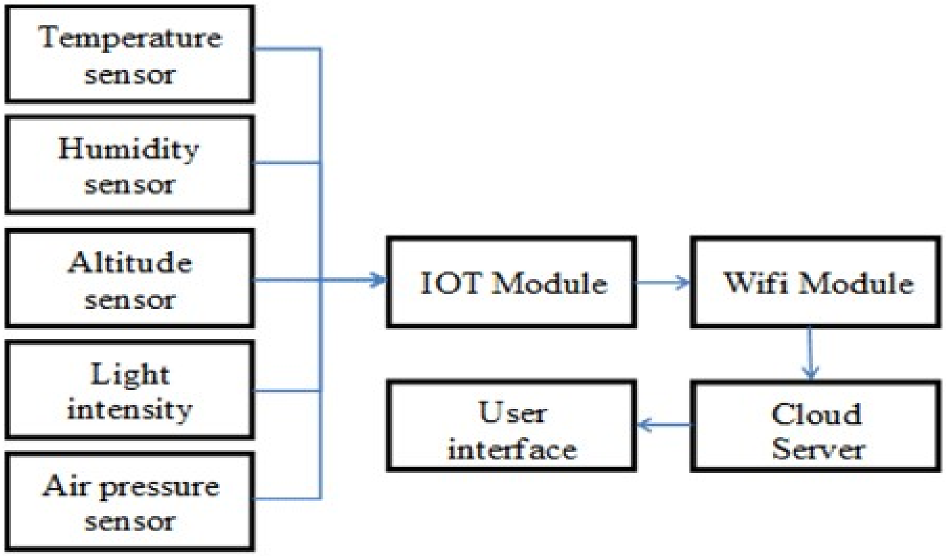


Figure 1.2: Block Diagram of a Typical IoT-Based System

1.3 OBJECTIVES AND OUTCOMES OF SMARTFUSION TRAINING

The SMARTFUSION program had these main objectives:

- To provide fundamental knowledge of AI, ML, and IoT concepts and industry applications.
- To enable hands-on experience with embedded systems, data acquisition, hardware interfacing, and cloud computing.
- To develop skills in software programming, machine learning model building, and dashboard creation.
- To complete mini-projects and a capstone project that applies learned skills in a real-world scenario

The outcomes included confidence in designing and deploying intelligent systems, proficiency in working with sensors and cloud tools, and improved teamwork and communication skills.

1.4 SOFTWARE AND HARDWARE TOOLS LEARNED

1.4.1 HARDWARE TOOLS

- **Development Boards:** Arduino Uno, ESP32, Raspberry Pi.
- **Sensors:** DHT11/22 (temperature/humidity), LDR (light), PIR (motion), soil moisture sensors.
- **Actuators & Devices:** Motors, relays, LED indicators.

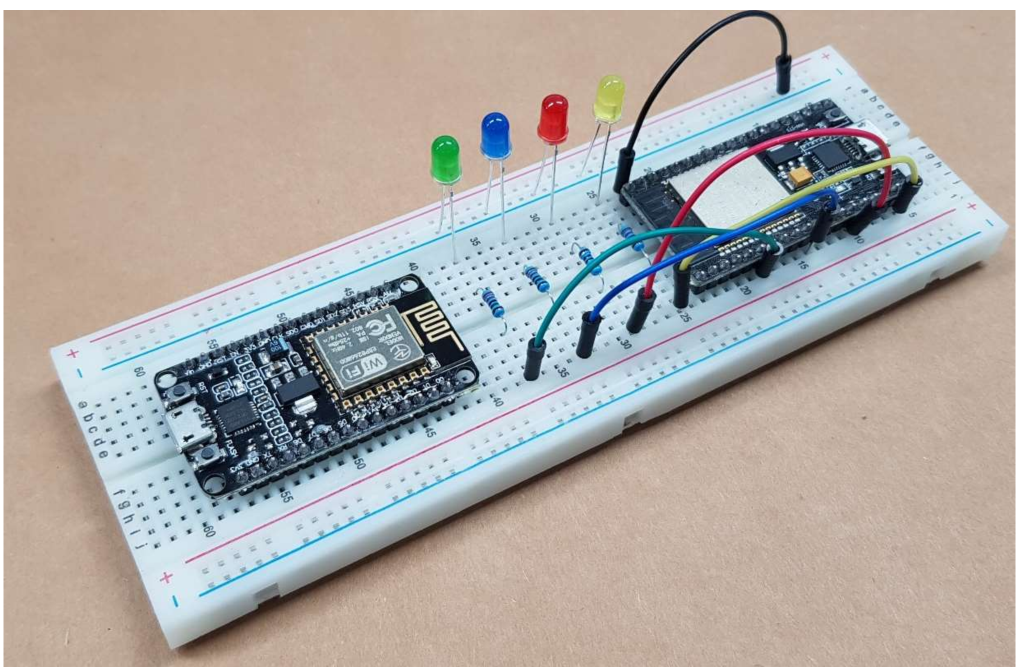


Figure 1.3: Example Hardware Image

1.4.2 SOFTWARE TOOLS

- **Programming Languages:** Python, Embedded C.
- **Data Analysis:** Jupyter Notebooks, Matplotlib, Seaborn.
- **IoT Platforms:** ThingSpeak, Azure IoT Hub, Blynk, MQTT Broker.
- **ML Libraries:** Scikit-learn, TensorFlow Lite (for microcontrollers).
- **Simulation Tools:** Arduino IDE, Tinkercad.

1.4.3 INTEGRATION WORKFLOW

- Wiring sensors to microcontrollers.
- Writing code for data acquisition and preprocessing.
- Cloud publishing and dashboard visualization.
- Creating, training, and deploying ML models on sensor data.
- Hardware Tools

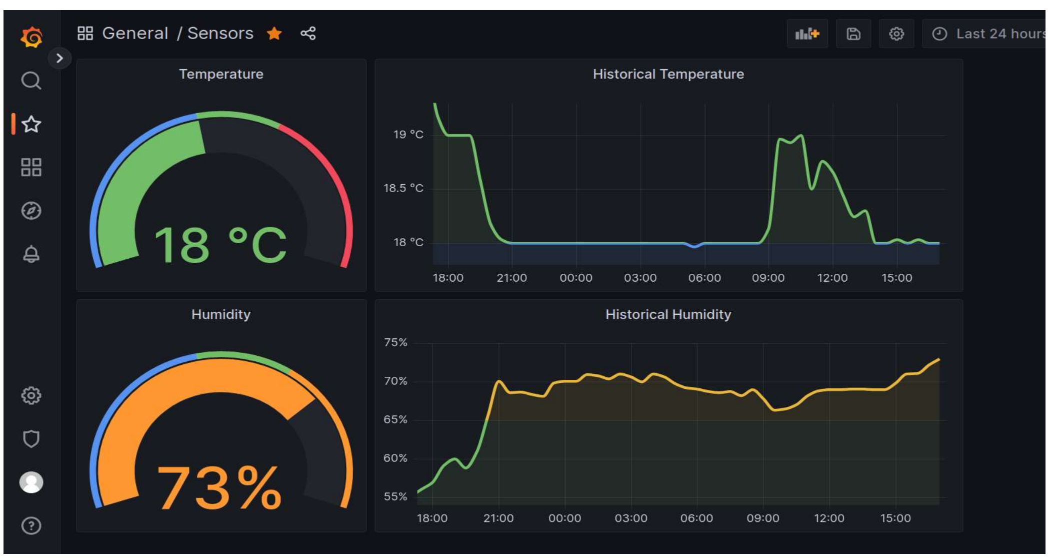


Figure 1.4: Screenshot of IoT Dashboard Displaying Real-Time

1.5 STRUCTURE AND FLOW OF THE TRAINING

Over four weeks, the training followed this flow:

Week 1: IoT architecture, board setup, basic coding**.**

Week 2: Sensor data acquisition, cloud storage, analytics.

Week 3: ML algorithm concepts, hands-on mini-projects in regression/classification.

Week 4: System integration and intelligent project deployment.

Students implemented hardware circuits, wrote scripts, analyzed sensor outputs, and presented mini-projects before beginning a larger group demonstration.

CHAPTER – 2

TRAINING WORK UNDERTAKEN

2.1 INTRODUCTION TO TRAINING

The SMARTFUSION four-week industrial training conducted at Guru Nanak Dev Engineering College, Ludhiana, aimed to develop both the theoretical understanding and practical expertise of students in the fields of Artificial Intelligence (AI), Machine Learning (ML), and Internet of Things (IoT). The training program was designed as a blend of structured lectures, simulator exercises, hardware experimentation, and mini-projects. These modules were strategically sequenced to ensure a gradual immersion from basic concepts into advanced system-building.

2.2 TOPICS COVERED DURING TRAINING

During the training, our group was exposed to several modern technologies and methods:

➢ **BASICS OF IOT & AI/ML:**

- Definition of IoT, its layered architecture (perception, network, edge, cloud, application), use cases in agriculture/industry.
- Introduction to AI types—reactive, limited memory—and their difference from ML.
- Types of ML (supervised/unsupervised), examples, and role in automation.

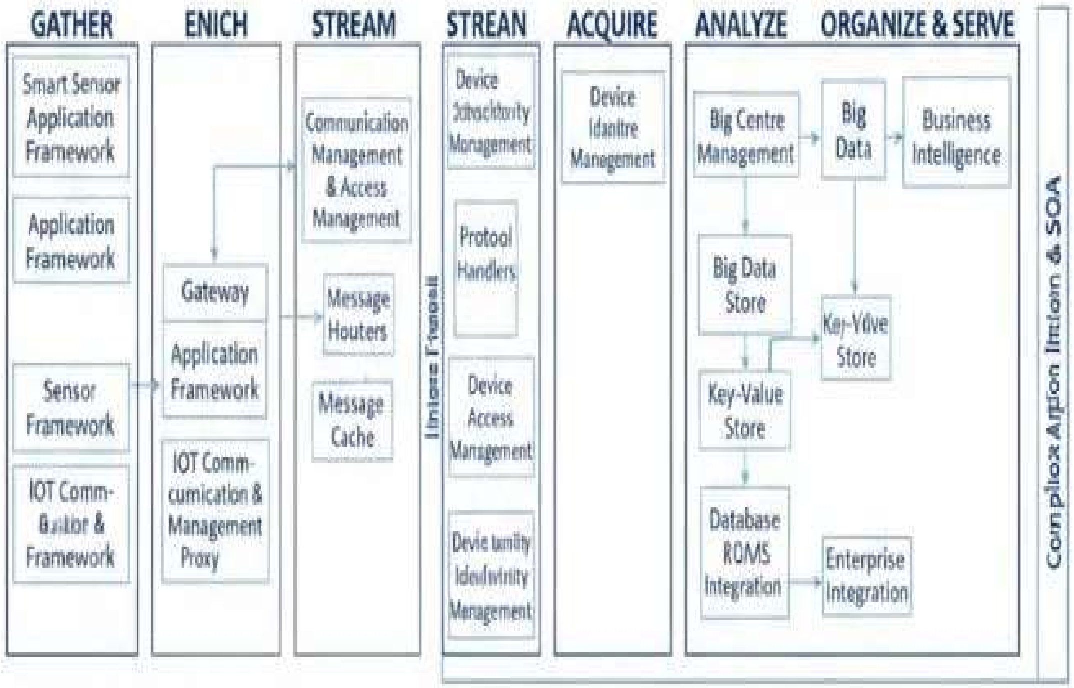
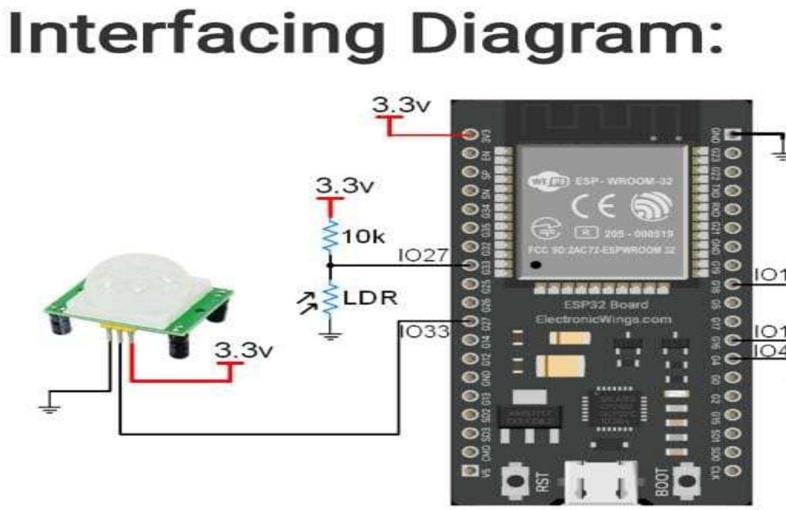


Figure 2.1 Diagram of IoT Architecture

➢ Sensor and Actuator Hardware:

- Deep dive into sensors: DHT11/22 (temperature/humidity), LDR (light), PIR (motion), soil moisture, real-world usage for data collection.
- DC gear motor theory, motor driver (L298N), and breadboard prototyping fundamentals.
- Working with ESP32/NodeMCU and Arduino Uno boards.



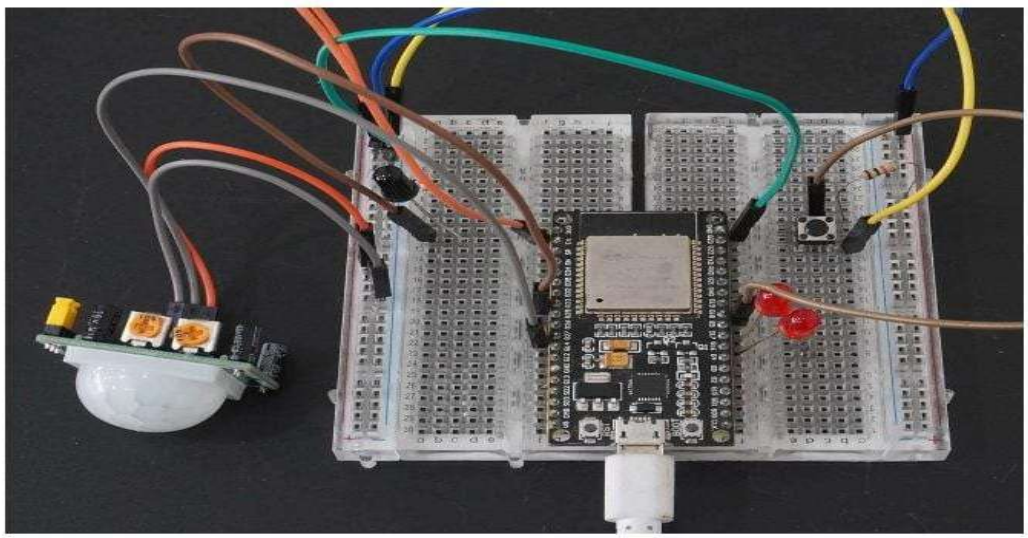


Figure 2.2 ESP32+Sensors

➢ Cloud, Connectivity, and Dashboarding:

- Setting up Wi-Fi communication on ESP32/NodeMCU.
- ThingSpeak, Virtuino, and Blynk apps for live sensor data monitoring.

- Data logging, channel creation, and custom widget design for dashboards.

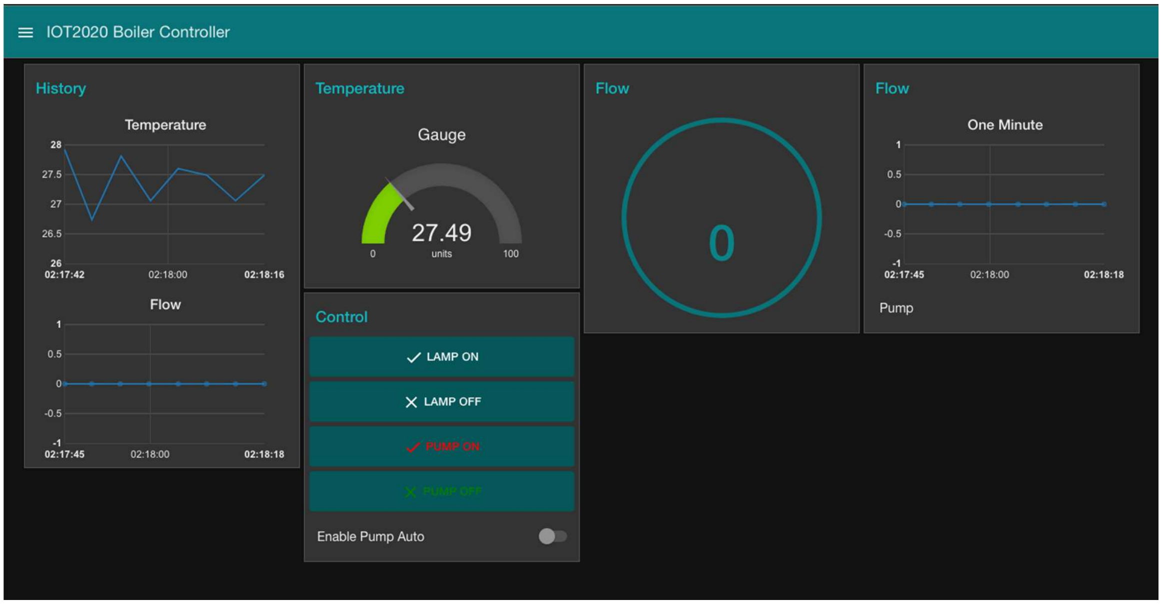


Figure 2.3 IoT Dashboard (Sample)

➢ Programming:

- Use of Arduino IDE and PlatformIO for hardware coding (C/C++ basic syntax, digital/analog I/O, sensor integration).
- Python basics (variables, control flow, functions) and libraries (Numpy, Pandas, Matplotlib, Scikit-learn) for analytics.

2.3 Practical Sessions: Day-by-Day Work

➢ **Week 1: Hardware and Connectivity**

- **Day 1–2:** Breadboard basics, power rails, digital pinout. Circuit assembly with ESP32 and DHT11

- **Day 3:** Coding the ESP32 for sensor data acquisition. Uploading first sensor reading to serial monitor.

- **Day 4–5:** Connecting LDRs and reading analog light values; controlling LED with sensor signal.

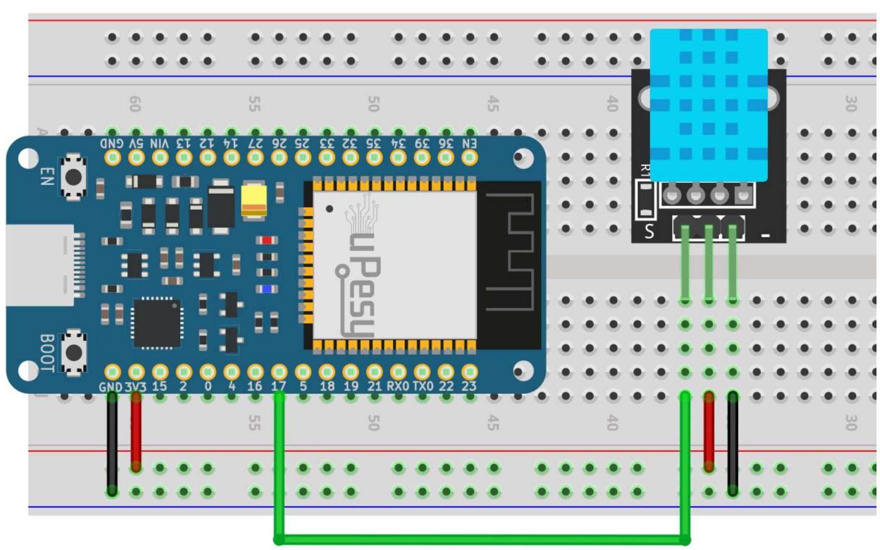


Figure 2.4 DHT11 Sensor on Breadboard

➢ **Week 2: Cloud and Data Visualization**

- **Day 1–2:** Registering on ThingSpeak/Blynk, creating a new channel/widget, connecting ESP32 to Wi-Fi.

- **Day 3:** Writing code to transmit sensor data (temp/humidity/light) to ThingSpeak at 2-minute intervals.
- **Day 4:** Analyzing ThingSpeak graphs, understanding IoT data latency.Day 5: Troubleshooting upload errors, dashboard customization.

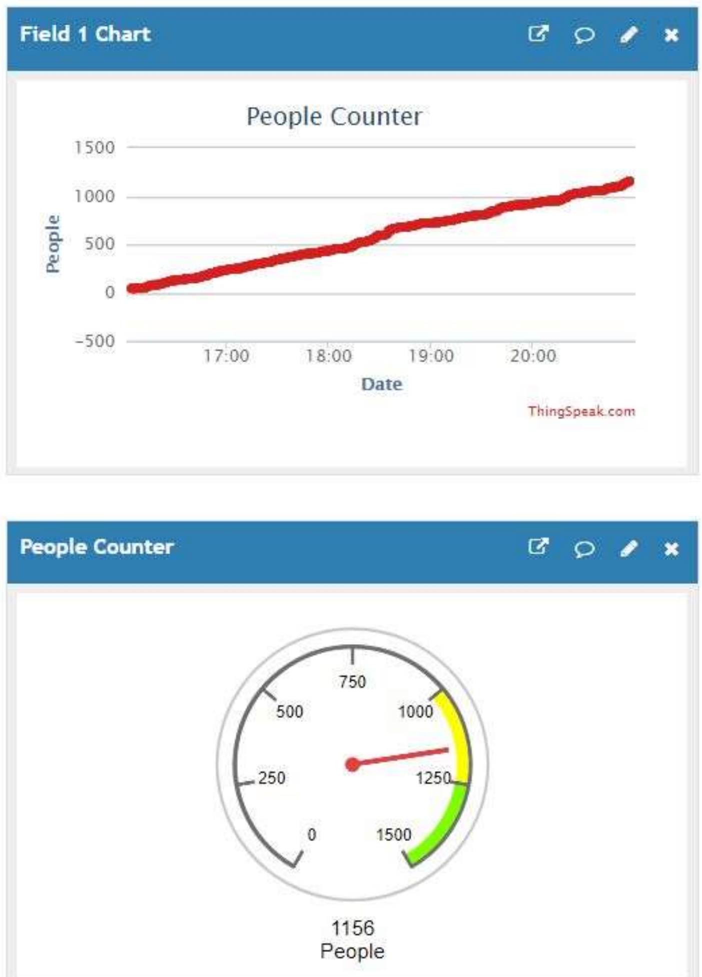


Figure 2.5 Thingspeak IoT Cloud Dashboard (MathWorks)

Week 2: Cloud & Connectivity

Mini Project 1 – Logging Sensor Data to Cloud Project

Title: ***Read Sensor Data and Log It to Cloud (ThingSpeak)***

Objective:

To design a system using ESP32 and DHT22 (temperature/humidity sensor) that collects live environmental data and uploads it to the ThingSpeak IoT cloud, where it can be accessed, stored, and visualized.

Components/Tools:

- ESP32 microcontroller (Wi-Fi enabled)
- DHT22 sensor
- Wokwi online simulator (for live circuit testing without hardware)
- ThingSpeak cloud platform
- Arduino IDE/C++ code

Steps:

- Connected ESP32 and DHT22 sensor on virtual breadboard via Wokwi.
- Used Arduino IDE with WiFi.h and DHT.h libraries.
- Code acquired temperature/humidity and uploaded every 15s to ThingSpeak using HTTP POST.
- Successfully displayed field graphs on the ThingSpeak dashboard in real time.

Sample Code Snippet:

```
#include <WiFi.h>

#include <DHT.h>

#include <HTTPClient.h>

#define DHTPIN 15

#define DHTTYPE DHT22

const char* ssid = "Wokwi-GUEST";
```

```
const char* password = "";

const char* apiKey = "YOUR_API_KEY";

const char* server = "http://api.thingspeak.com/update";

DHT dht(DHTPIN, DHTTYPE);

void setup() { ... }

void loop() { ... }
```

- **Week 3: Data Analytics and Machine Learning**

- **Day 1–2:** Preprocessing data using Python; cleaning, normalizing, plotting with Matplotlib.

- **Day 3:** Introduction to regression, training basic ML model with scikit-learn, testing model on collected temperature data.

- **Day 4:** Classification task: anomaly detection in sensor values.

- **Day 5:** Mini-projects coded—Team A: light prediction model; Team B: humidity-based warning system.

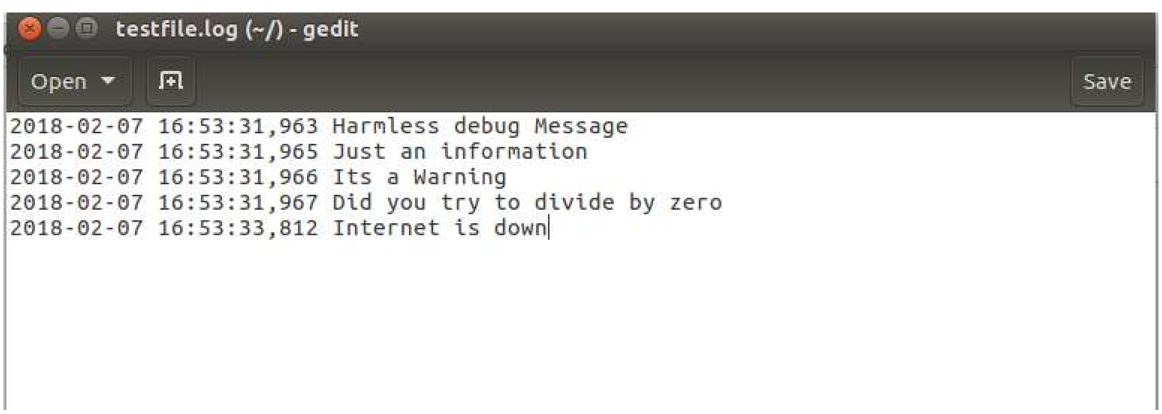


Figure2.6Python Data Logging Example

- **Week 4: Mini Project 2 – Real-Time Custom IoT Dashboard with Live Graphs Project**

Title: Real-Time IoT Dashboard with Graphs using ESP32, DHT22, Wokwi, ThingSpeak, and HTML/JavaScript

Objective:

To create a working real-time system that reads temperature and humidity from DHT22 (via ESP32), transmits data to ThingSpeak, and visualizes it using custom HTML/CSS/JS dashboard for live graphs.

Components/Tools:

- ESP32
- DHT22 Sensor
- Wokwi Simulator
- ThingSpeak platform
- HTML, CSS, JavaScript for custom front-end
- CodePen for live web prototype

Steps Followed:

1. Designed virtual circuit in Wokwi.
2. Programmed ESP32 with Arduino code to connect to Wi-Fi, read DHT22 values, and send them to ThingSpeak HTTP API.
3. Built a live web dashboard using HTML and JS, fetching ThingSpeak data via API and plotting with live updating charts.

CHAPTER- 3

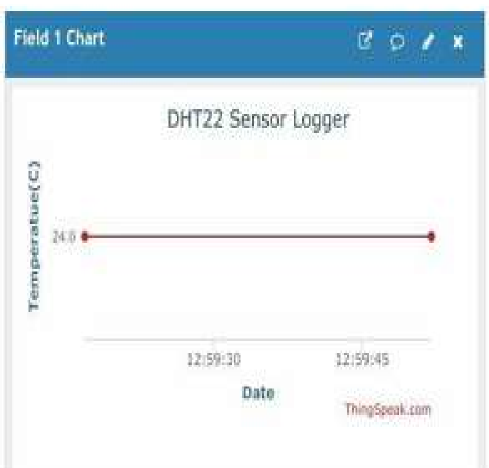
RESULTS AND DECISION

3.1RESULTS

3.1.1 MINI PROJECT 1: CLOUD LOGGING OF SENSOR DATA

The first mini project was successfully implemented using ESP32 and DHT22 to collect environmental data, which was uploaded to ThingSpeak at regular intervals.

- **Graphical Results:** Real-time temperature and humidity values appeared as separate fields on the ThingSpeak channel dashboard, both in tabular and graph formats.



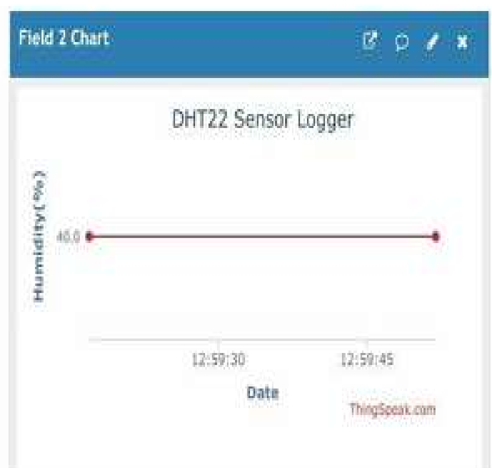


Figure 3.1 Humidity Graph from same channel

- **Snapshot:**

Code output showed “Connecting to WiFi… WiFi connected! Data sent to ThingSpeak!” confirming end-to-end communication.

- **Observation:**

The data matched the expected readings from the physical environment (lab/classroom), validating sensor calibration and cloud integration.

3.1.2 Mini Project 2: Real-Time IoT Dashboard

The second mini project created a live dashboard using HTML, CSS, and JavaScript, which fetched and visualized live sensor data from ThingSpeak**.**

Dashboard View: Custom web dashboard (tested on CodePen) successfully fetched fields via API and plotted live-updating graphs for temperature and humidity.

Figure 3.2 IoT Dashboard Plottting Data

- **Results :**

Immediate graph updates with each new upload from the ESP32 (every 15 seconds).Displayed historical trends and live statistics (min, max, avg readings shown beside charts).

- **Observation:**

The system demonstrated the feasibility of remote monitoring and analytics, foundational for any real IoT deployment.

➢ **CRITICAL DISCUSSION AND OBSERVATIONS**

- **Reliability:**

  Both mini-project systems ran stably for hours, proving the robustness of code and hardware integration.

- **Challenges:**

  Occasional WiFi connectivity drops were handled by retry logic in code, ensuring minimal data loss.

- **Data Quality:**

  Some sensor values fluctuated due to environmental changes (e.g., open window/lab traffic)—enriching understanding of real-world edge cases.

- **Teamwork:**
- Split responsibilities (coding, debugging, cloud dashboarding) enabled timely completion and peer learning.
- **Innovation:**

The move to a full browser dashboard for live reading was realized using only the training content—students appreciated the instant feedback and industry relevance.

- **LEARNING OUTCOMES FROM TRAINING TOOLS**

- Became proficient with ESP32, DHT22, Arduino IDE, and Wokwi simulator for virtual prototyping and debugging.
- Developed working knowledge of cloud IoT APIs (ThingSpeak), HTTP protocols, and data visualization.
- Improved programming (C++, Python, JS), circuit design, and team presentation skills.
- Developed troubleshooting approaches for hardware, software, and network issues.
- Understood full IoT pipeline: sensor → microcontroller → WiFi → cloud → live dashboard.

- **Project Development – AgroSolAR Automated Rotation**

The knowledge and skills developed in the previous modules were applied to build the **AgroSolAR Automated Rotation** system.

- **IoT and Sensor Integration:**ESP32, DHT11, LDRs, L298N, DC gear motors fully wired and debugged.
- Continuous data push to the cloud backend for tracking and analytics.

:

Figure 3.3: Team's AgroSolAR Hardware Setup.

- **ML Model Creation:**
  - Linear regression model coded in Python to predict optimal panel tilt.
  - Real-time correction based on sunlight and temperature sensor trends.
  - Smoothed motor control logic to prevent over-rotation.

- **Integration of IoT and Cloud Computing**
  - Continuous recording of environmental values and solar angle on public dashboard.
  - Real-time monitoring for faculty and remote demonstration during presentation.
    - Multiple graphs: sunlight vs panel angle, temperature/humidity vs time

3.2 DECISION

Based on the dashboard data, graphs, and observed performance: The solar tracking system performs efficiently and is able to increase energy generation compared to static solar panels. The integration of IoT allows remote monitoring, which is useful for agricultural management. The use of AI and ML improves tracking accuracy, making the system adaptive to changing sunlight conditions. The system is feasible and practical for agricultural applications, such as powering irrigation systems, sensors, and other smart-farming devices.

Decision: The AgroSolAR project is successful, effective, and suitable for real-world agricultural use, combining renewable energy, IoT, and AI technologies for smart farming solutions.

CHAPTER 4

CONCLUSION

4.1 CONCLUSION

The SmartFusion industrial training has provided a practical and theoretical foundation in the most in-demand technologies: Artificial Intelligence, Machine Learning, and the Internet of Things. Over the course of four weeks, each session and mini-project contributed to a deepened understanding of both hardware and software essentials. Key conclusions from the training and mini-projects are as follows:

- The process of reading sensor data, uploading it to the cloud, and visualizing real-time results on dashboards enabled the team to practice the whole IoT pipeline in a hands-on way.
- Working with ESP32, DHT22 sensors, AWS/ThingSpeak/Blynk cloud platforms, and Python/JavaScript for analytics made us more comfortable with industry tools and programming.
- Mini-projects, like cloud logging and web dashboard visualization, allowed each member to build technical confidence and gain experience with typical real-world challenges (e.g., noisy sensor data, Wi-Fi dropout, teamwork).
- As a group, we developed not only deep technical knowledge but also collaboration, troubleshooting, and communication skills—vital qualities for any successful Engineer.This training has encouraged further exploration in IoT and AI, and inspired new ideas for smart projects in our academic and professional futures.

REFERENCES

[1]S. MathWorks, "ThingSpeak Documentation & Tutorials," Available: https://thingspeak.com/docs [Online]. Accessed: Oct. 15, 2025.

[2]Wokwi Inc., "Online ESP32 and Sensor Simulator," Wokwi.com, [Online]. Available: https://wokwi.com/. Accessed: Oct. 15, 2025.

[3]ElectronicWings, "DHT22 Sensor Interfacing with ESP32/Arduino Boards," [Online]. Available:https://www.electronicwings.com/nodemcu/dht11-interfacing-with-nodemcu. Accessed: Oct. 15, 2025.

[4]CodePen Community, "Real-Time IoT Dashboard Examples," CodePen.io, [Online]. Available: https://codepen.io/. Accessed: Oct. 15, 2025.

[5] S. MathWorks, "Create Custom ThingSpeak Dashboard Apps," [Online]. Available: https://www.mathworks.com/help/thingspeak/create-custom-thinghttp-apps.html. Accessed: Oct. 15, 2025.