National Textile University, Faisalabad

Department of Computer Science

Name:	Muhammad Hassaan Raza
Class:	BSAI-6th
Registration No:	22-NTU-CS-1362
Assignment	Lab13-Lab/Home tasks
Course Code:	AIE-3079
Course Name:	Internet of Things Fundamentals
Submitted To:	Mr. Nasir Mehmood
Submission Date:	20-May-2025

LAB 13 REPORT

TASK 1: ARDUINO MQTT PUBLISHER FOR DHT SENSOR

What I Did:

I used Arduino code to send temperature and humidity data from a DHT11 sensor to an MQTT broker (Mosquitto). The ESP32 connects to Wi-Fi and publishes data every second.

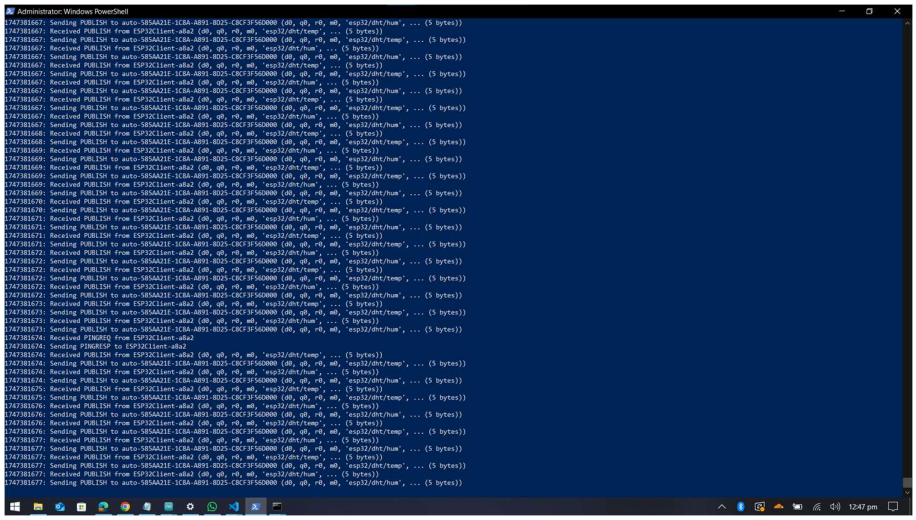
Results:

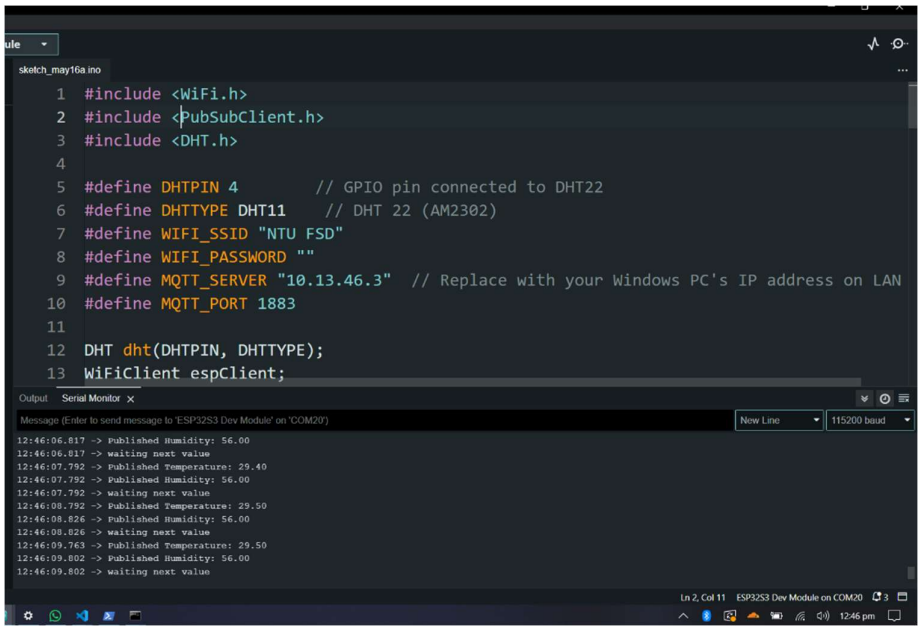
- The Serial Monitor showed live updates like:
  Published Temperature: 29.40°C
  Published Humidity: 56.00%
- The data was sent successfully to the topic esp32/dht/temp and esp32/dht/hum.

Problems Faced:

- Sometimes, the sensor failed to read values, but the code handled it by printing an error.

Screenshot Attached:





TASK 2: STORE MQTT DATA IN INFLUXDB

What I Did:

I ran a Python script (1-dht_data_only.py) to subscribe to MQTT topics and save the sensor data to InfluxDB.

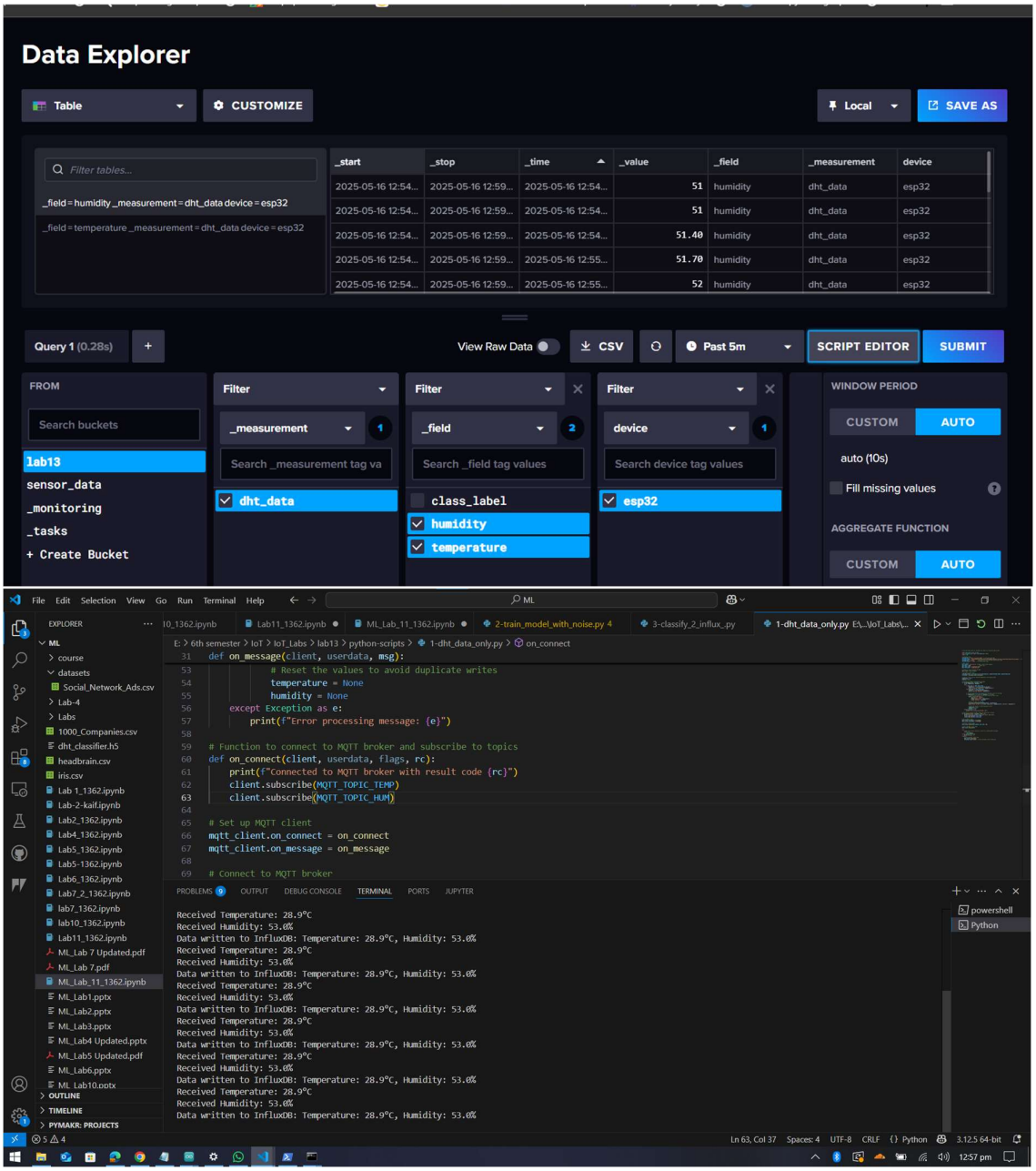
Results:

- The script printed confirmations like:
  Received Temperature: 29.40°C
  Data written to InfluxDB: Temperature: 29.40°C, Humidity: 56.00%
- Data was stored in InfluxDB under the measurement dht_data with tags and fields.

Problems Faced:

- First, the script didn't connect because of a wrong IP address. I fixed it by checking the broker address.

screenshot Attached:



TASK 3: TRAIN MODEL WITH NOISE

What I Did:

I ran 2-train_model_with_noise.py to create a synthetic dataset and train a neural network to classify temperature/humidity into 5 classes (e.g., "Hot and Dry").

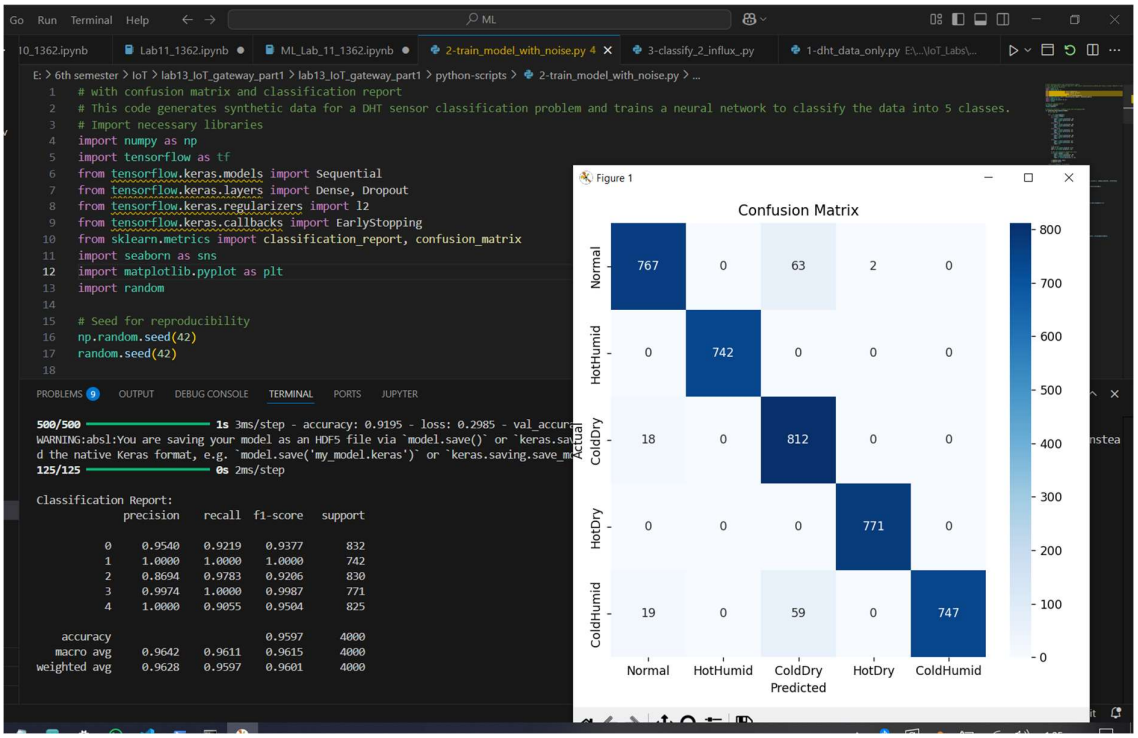
Results:

- The model achieved **~92% accuracy** .
- The confusion matrix showed good performance but some confusion between similar classes (e.g., "Normal" and "Cold and Humid").

Problems Faced:

- The training was slow on my laptop, but eventually it trained.

Screenshots Attached:



TASK 4: CLASSIFY AND STORE IN INFLUXDB

What I Did:

I executed 3-classify_2_influx.py to read live sensor data, classify it using the trained model, and save everything (temperature, humidity, and class label) to InfluxDB.

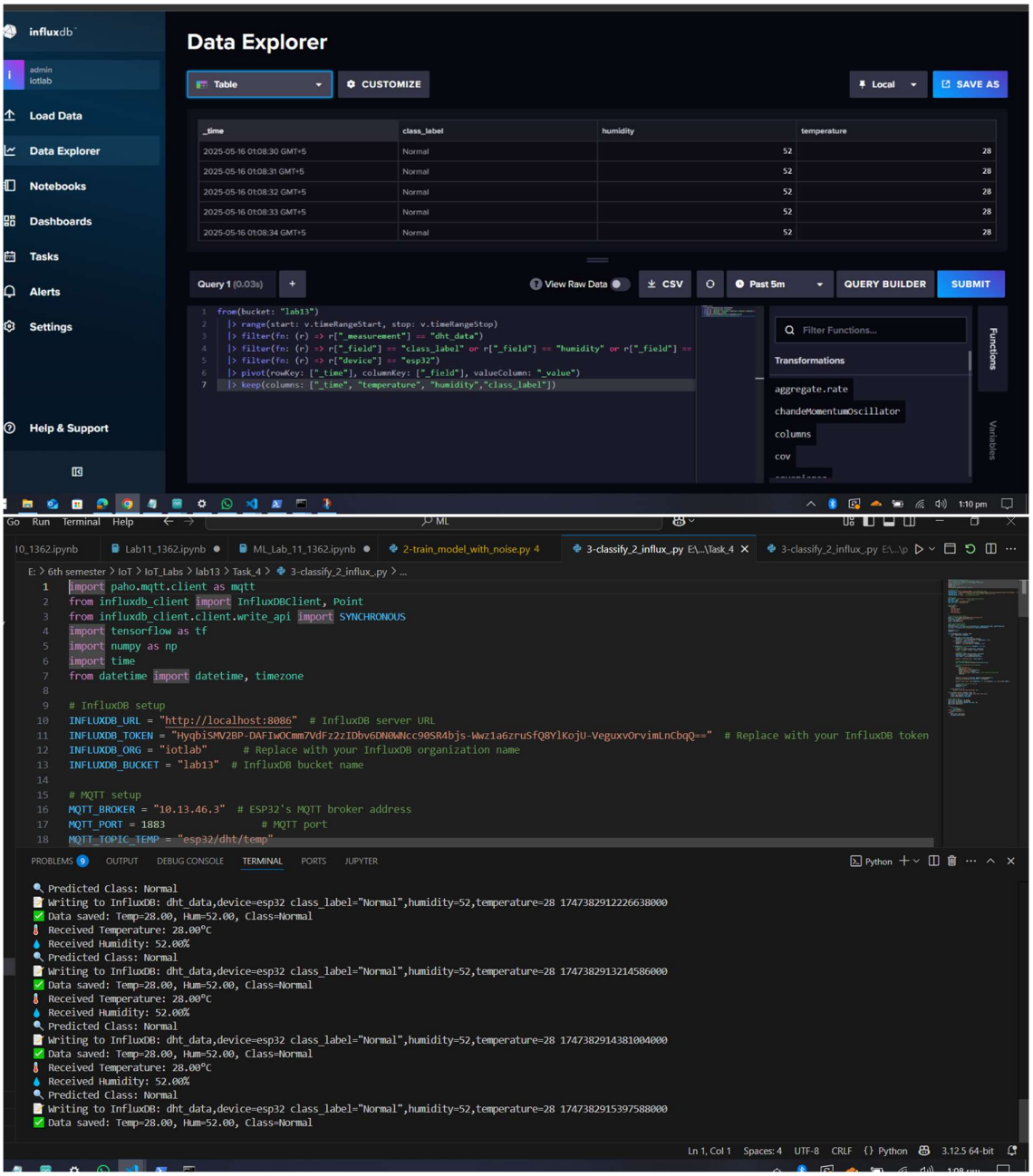
Results:

- The console showed real-time updates:
  🌡️ Received Temperature: 29.40°C
  🔍 Predicted Class: Normal
  ✅ Data saved to InfluxDB
- The InfluxDB dashboard displayed all fields: temperature, humidity, and class_label.

Problems Faced:

- At first, the class labels didn't appear in InfluxDB because of a typo in the field name. I fixed it by checking the script.

Screenshots Attached:


influxdb
admin
iotlab
Load Data
Data Explorer
Notebooks
Dashboards
Tasks
Alerts
Settings
Help & Support
Data Explorer
Table
CUSTOMIZE
Local
SAVE AS
_time
class_label
humidity
temperature
2025-05-16 01:08:30 GMT+5
Normal
52
28
2025-05-16 01:08:31 GMT+5
Normal
52
28
2025-05-16 01:08:32 GMT+5
Normal
52
28
2025-05-16 01:08:33 GMT+5
Normal
52
28
2025-05-16 01:08:34 GMT+5
Normal
52
28
Query 1 (0.03s)
View Raw Data
CSV
Past 5m
QUERY BUILDER
SUBMIT
from(bucket: "lab13")
|> range(start: v.timeRangeStart, stop: v.timeRangeStop)
|> filter(fn: (r) => r["_measurement"] == "dht_data")
|> filter(fn: (r) => r["_field"] == "class_label" or r["_field"] == "humidity" or r["_field"] ==
|> filter(fn: (r) => r["device"] == "esp32")
|> pivot(rowKey: ["_time"], columnKey: ["_field"], valueColumn: "_value")
|> keep(columns: ["_time", "temperature", "humidity","class_label"])
Filter Functions...
Transformations
aggregate.rate
chandeMomentumOscillator
columns
cov
Functions
Variables
1:10 pm

Go Run Terminal Help
ML
10_1362.ipynb
Lab11_1362.ipynb
ML_Lab_11_1362.ipynb
2-train_model_with_noise.py 4
3-classify_2_influx_.py E\...\Task_4
3-classify_2_influx_.py E\...\p
E: > 6th semester > IoT > IoT_Labs > lab13 > Task_4 > 3-classify_2_influx_.py > ...
import paho.mqtt.client as mqtt
from influxdb_client import InfluxDBClient, Point
from influxdb_client.client.write_api import SYNCHRONOUS
import tensorflow as tf
import numpy as np
import time
from datetime import datetime, timezone
# InfluxDB setup
INFLUXDB_URL = "http://localhost:8086" # InfluxDB server URL
INFLUXDB_TOKEN = "HyqbiSMV2BP-DAFIwOCmm7VdFz2zIDbv6DN0WNcc90SR4bjs-Wwz1a6zruSfQ8YlKojU-VeguxvOrvimLnCbqQ==" # Replace with your InfluxDB token
INFLUXDB_ORG = "iotlab" # Replace with your InfluxDB organization name
INFLUXDB_BUCKET = "lab13" # InfluxDB bucket name
# MQTT setup
MQTT_BROKER = "10.13.46.3" # ESP32's MQTT broker address
MQTT_PORT = 1883 # MQTT port
MQTT_TOPIC_TEMP = "esp32/dht/temp"
PROBLEMS 9
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
Python
Predicted Class: Normal
Writing to InfluxDB: dht_data,device=esp32 class_label="Normal",humidity=52,temperature=28 1747382912226638000
Data saved: Temp=28.00, Hum=52.00, Class=Normal
Received Temperature: 28.00°C
Received Humidity: 52.00%
Predicted Class: Normal
Writing to InfluxDB: dht_data,device=esp32 class_label="Normal",humidity=52,temperature=28 1747382913214586000
Data saved: Temp=28.00, Hum=52.00, Class=Normal
Received Temperature: 28.00°C
Received Humidity: 52.00%
Predicted Class: Normal
Writing to InfluxDB: dht_data,device=esp32 class_label="Normal",humidity=52,temperature=28 1747382914381004000
Data saved: Temp=28.00, Hum=52.00, Class=Normal
Received Temperature: 28.00°C
Received Humidity: 52.00%
Predicted Class: Normal
Writing to InfluxDB: dht_data,device=esp32 class_label="Normal",humidity=52,temperature=28 1747382915397588000
Data saved: Temp=28.00, Hum=52.00, Class=Normal
Ln 1, Col 1
Spaces: 4
UTF-8
CRLF
Python
3.12.5 64-bit