

1

Name: Fathima Sania

SUID: 518689600

Netid: fsania@syr.edu

Assignment Report

1- Steps Taken- Step 1: Created a ThingSpeak account and set up a new channel called Weather Data - Week 12.

Step 2: Configured four data fields:

- Field 1: Temperature
- Field 2: Humidity
- Field 3: Pressure
- Field 4: Wind Speed

Step 3: Registered for OpenWeatherMap and obtained my API key.

Step 4: Wrote a Python script to fetch live weather data (Temperature, Humidity, Pressure, Wind Speed) from OpenWeatherMap API and send it to ThingSpeak.

Step 5: Ran the Python script on my computer (Visual Studio Code + Terminal), and verified successful data updates in the ThingSpeak channel.

Step 6: Created visualizations in ThingSpeak for all four fields to monitor real-time weather data.

Step 7: Took screenshots of the running script and ThingSpeak visualizations for submission.

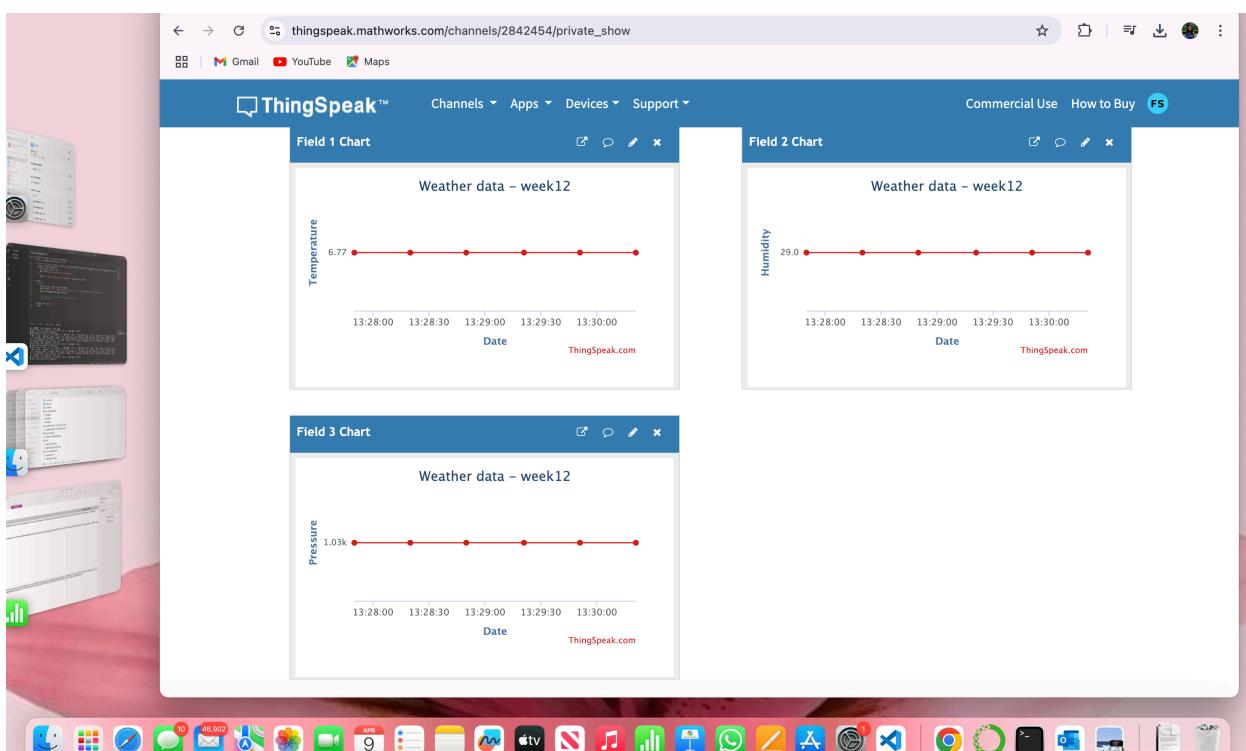
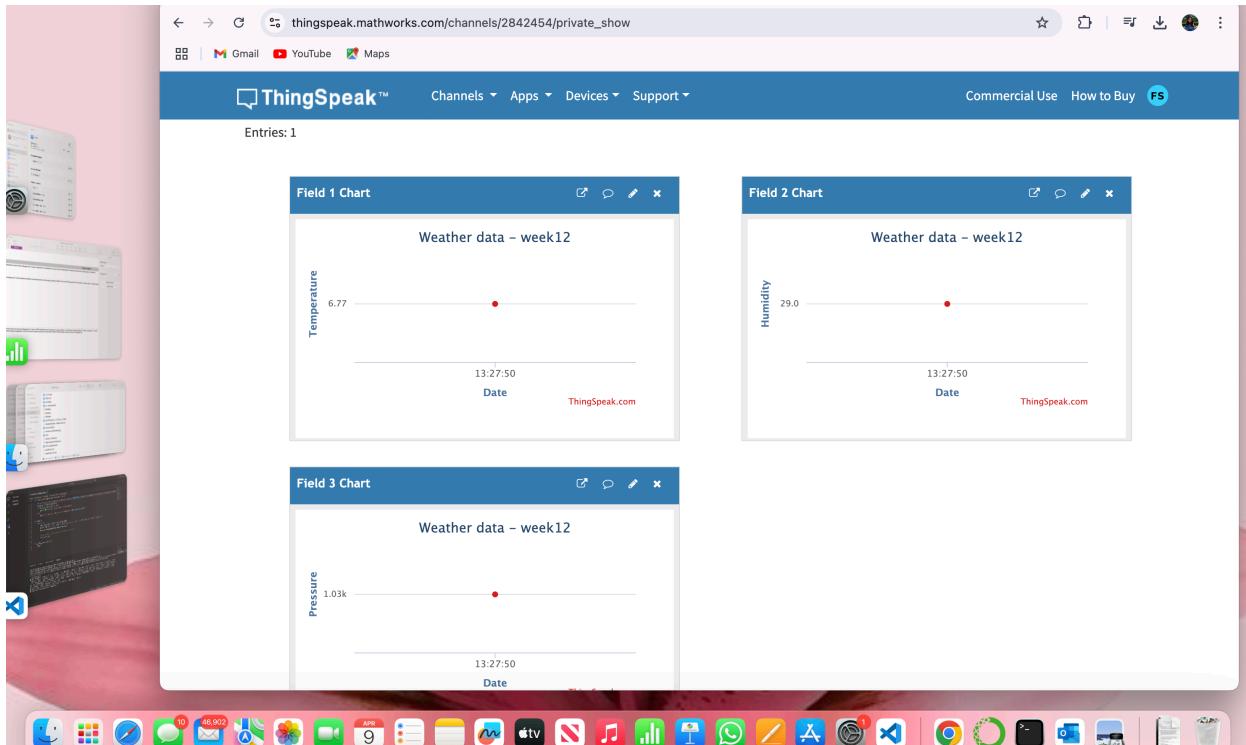
2

Name: Fathima Sania

SUID: 518689600

Netid: fsania@syr.edu

Screenshots of the temperature, humidity, pressure and wind speed changing with time and I also added the location channel as NYC to give it more depth.

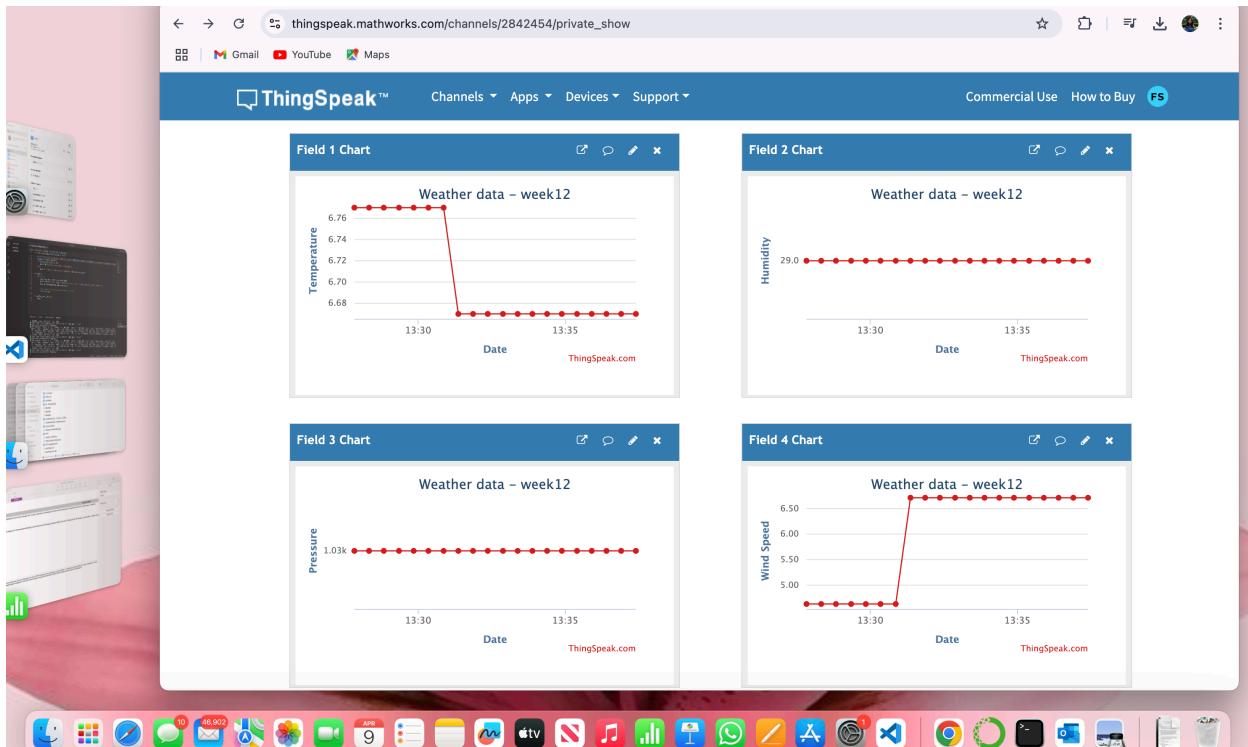
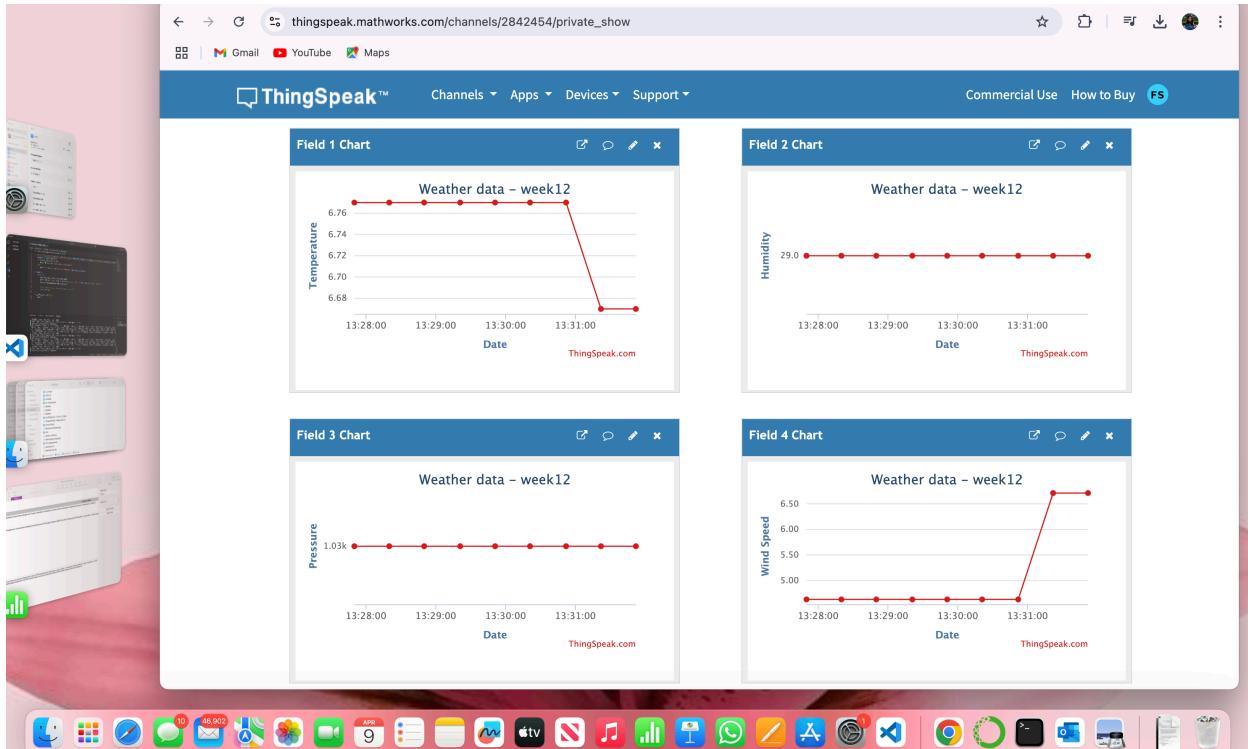


3

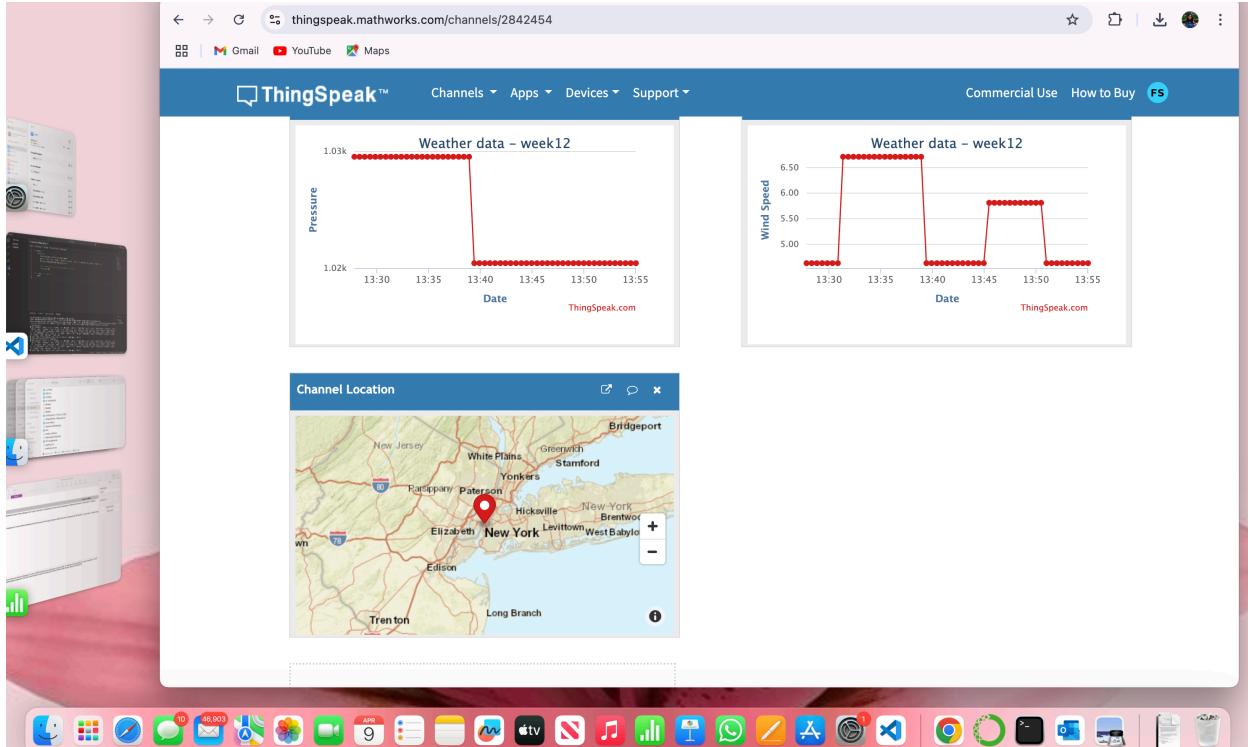
Name: Fathima Sania

SUID: 518689600

Netid: fsania@syr.edu



Name: Fathima Sania
 SUID: 518689600
 Netid: fsania@syr.edu



The code that I ran in VS-Code

```

import time
# Your API Keys
OPENWEATHERMAP_API_KEY = 'eac7476519afaf1eb143b12859c197af'
THINGSPEAK_API_KEY = 'IN08NTFLJ5ZK033'

# Location for weather data
CITY = 'New York'

def get_weather_data():
    url = f'http://api.openweathermap.org/data/2.5/weather?q={CITY}&appid={OPENWEATHERMAP_API_KEY}&units=metric'
    response = requests.get(url)
    data = response.json()

    print(" API response:", data)

    # Check if API call was successful
    if response.status_code != 200:
        print(f" Error fetching weather data: {data.get('message', 'Unknown error')}")
        return None, None, None

    try:
        # Extract required data
        temperature = data['main']['temp']
        humidity = data['main']['humidity']
        pressure = data['main']['pressure']

        return temperature, humidity, pressure
    except:
        print(" Error extracting data from API response")
        return None, None, None

```

The screenshot shows the VS-Code interface with the "weather-to-thingspeak.py" file open in the editor. The code uses the `requests` library to fetch weather data from OpenWeatherMap for the city of New York. It then sends the data to ThingSpeak using its API. The code includes error handling for API calls and data extraction.

Name: Fathima Sania
SUID: 518689600
Netid: fsania@syr.edu

3. Reflection

This assignment was an exciting experience because I got to integrate real-time weather data from an external API and visualize it on the ThingSpeak platform. Initially, I faced some challenges with running the script and understanding the data flow between APIs and ThingSpeak, but debugging helped me learn more about API responses and data handling in Python. Watching the data update live and seeing the visualizations change was a rewarding moment. Overall, this exercise gave me hands-on experience with IoT data visualization and improved my confidence in working with APIs and cloud platforms.