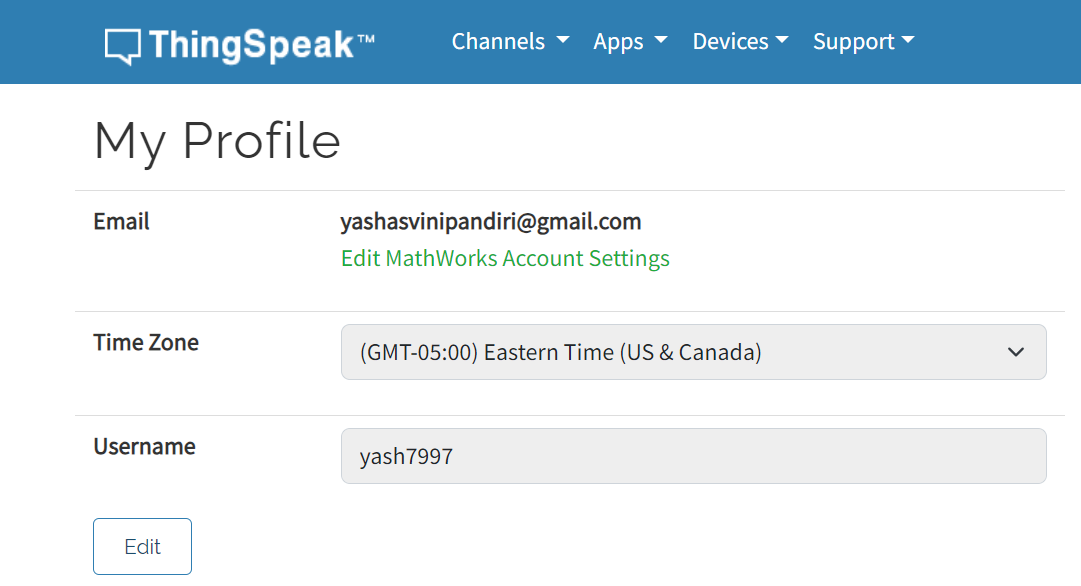
# Steps that I have used in developing the IOT system:

* Setup an account on IoT platform:
  + The first step was to set up an account on the IoT platform. My choice was ThingSpeak as it is very lightweight, and I already have some experience using it.



* Create 2 channels:
  + Created 2 channels on the ThingSpeak website. Each channel has a unique ID and acts as the environmental station. Both channels have 3 fields namely Temperature, Humidity and Co2 values. Also, I updated the channel visibility to public.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Create MQTT devices:
  + The protocol to be used in communication is MQTT. Hence, I created 2 MQTT devices and added one channel to each of these devices. Also downloaded the credentials with username and password for each device.

A screenshot of a computer

Description automatically generated

* Python script to generate random values for sensor data:
  + Using random library of python, I generated 3 random values for temperature, humidity and Co2 levels within the range specified in the assignment.
* Publish data onto ThingSpeak using IoT:
  + Randomly generated data is published on each of the channels using corresponding MQTT device. Python’s paho.mqtt.publish is utilized to connect over to the ThingSpeak website by specifying the correct authentication details.

A screenshot of a computer

Description automatically generated

* Export the CSV data:
  + After running the program for a certain period, the data displayed on the ThingSpeak website can be exported to a CSV file for visualization purposes using matplotlib on Python.
* Visualize using Matplotlib:
  + Using Python’s matplotlib library, the generated CSV’s can be read into data frames with unique ID for each channel data. The combined data can then be visualized using the appropriate filters such as recent data, all channels etc.
* Push to GitHub:
  + Save both the python scripts, CSV files and upload them along with a Readme file onto the GitHub repository. Committed the screenshots as well onto the GitHub repository.

A screenshot of a computer

Description automatically generated

**# GitHub repository: https://github.com/YashasviniP/IoT\_Assignment3.git**

# Screen shots of your output

#All fields from Station 1

* Fig 1 – Temperature vs Time for Station 1
* Fig 2 – Humidity vs Time for Station 1
* Fig 3 – Co2 vs Time for Station 1
* Fig 4 – All Fields vs Time for Station 1

A group of graphs with different colored lines

Description automatically generated with medium confidence

#Temperature field from both stations

* Fig 1 – Temperature vs Time for Station 1
* Fig 2 – Temperature vs Time for Station 2

A green and blue lines

Description automatically generated

Note: The figures generated above are for a small amount of data as Graph was very densely populated for 5hrs data. The Python code for visualization plots exactly latest 5hrs data. The data is available on ThingSpeak and can be exported to a CSV file.