

# AI Chatbot for PM 2.5 and Climate Awareness

โดย

นางสาวณัฏฐ์รพี แสงสมบูรณ์ 6714650442

เสนอ

รศ.ดร.พงศ์ประพันธ์ พงษ์โสภณ

รายวิชา Nature of STEM 01159513  
ภาควิชาการศึกษา คณะศึกษาศาสตร์  
มหาวิทยาลัยเกษตรศาสตร์

## 1. ชื่อโครงการ: AI Chatbot for PM 2.5 and Climate Awareness

รายวิชาและภาคการศึกษา: 01159513 Nature of STEM , ภาคเรียนที่ 2/2567

โดย นางสาวณัฏฐ์พี แสงสมบูรณ์ (รหัสนิสิต: 6714650442)

อาจารย์ผู้สอน: รศ.ดร.พงศ์ประพันธ์ พงษ์โสภณ

วันที่ : 12 มีนาคม 2025

## 2. บทนำและความเป็นมา (Introduction & Background)

### บริบทและเหตุผล

สาเหตุที่เลือกด้านสิ่งแวดล้อมเนื่องจากได้ทำการวิเคราะห์จากสภาพปัญหาใกล้ตัวของตนเองก่อนเป็นอันดับแรกว่าโรงเรียนที่ตนเองสอนอยู่นั้นประสบปัญหาสิ่งใด สิ่งที่พบ คือ นักเรียนจำนวนมากประสบต่อความเสี่ยงในเรื่องฝุ่นควันมลพิษทางอากาศอยู่เสมอ เนื่องจากบริบทของโรงเรียนตั้งอยู่ในเขตนิคมโรงงานอุตสาหกรรม ในบางวันจะมีกลุ่มควันสีขาวที่มาพร้อมกับกลิ่นและทำให้แสบจมูก นักเรียนไม่สามารถจำแนกได้ว่าในวันดังกล่าวนั้นเป็นมลพิษจากโรงงานอุตสาหกรรมบริเวณรอบโรงเรียน ฝุ่นจาก PM 2.5 หรือเป็นผลมาจากกลุ่มหมอกในช่วงเช้า คงจะดีไม่น้อยหากมีเครื่องมืออะไรบางอย่างที่เราสามารถสอบถามและโต้ตอบได้ในขณะนั้นเลยว่าสภาพอากาศที่กำลังเผชิญนั้นคืออะไร เป็นอันตรายต่อสุขภาพของเราหรือไม่ นิสิตจึงอยากพัฒนาเครื่องมือที่ช่วยอำนวยความสะดวกแก่ผู้เรียนในการตรวจเช็คคุณภาพอากาศ จึงได้พัฒนาการเขียน python ที่สามารถระบุคุณภาพอากาศแบบเรียลไทม์ เพื่อสามารถแจ้งเตือนข้อมูลให้กับนักเรียนได้ และสามารถนำเซตบทนั้นมาเป็นเครื่องมือประกอบการเรียนการสอนเรื่องภาวะโลกร้อนในบทเรียนชั้นประถมศึกษาตอนปลาย

การนำโค้ดและเครื่องมือดิจิทัลมาประยุกต์ใช้ในการสอนสิ่งแวดล้อมในศตวรรษที่ 21 มีความสำคัญอย่างยิ่ง เนื่องจากกระบวนการเรียนรู้ในปัจจุบันเกิดขึ้นอย่างรวดเร็ว ผู้เรียนสามารถเข้าถึงข้อมูลผ่านเครื่องมืออิเล็กทรอนิกส์ต่าง ๆ ได้โดยไม่จำกัดอยู่แค่ตำราเรียน นอกจากนี้ ทักษะด้านเทคโนโลยีและการวิเคราะห์ข้อมูลยังเป็นที่ต้องการในตลาดแรงงาน ทำให้การเรียนรู้สิ่งแวดล้อมควบคู่ไปกับการใช้โค้ดและดิจิทัลช่วยเสริมสร้างความเข้าใจเชิงลึก และพัฒนาทักษะที่จำเป็นสำหรับอนาคต

### 3. วัตถุประสงค์และผลลัพธ์การเรียนรู้ (Objectives and Learning Outcomes)

#### 1. วัตถุประสงค์ของโครงการ

- พัฒนาโปรแกรม Python เพื่อให้นักเรียนวิเคราะห์คุณภาพอากาศแบบเรียลไทม์
- นักเรียนสามารถสืบเสาะหาความรู้เรื่อง PM 2.5 และภาวะโลกร้อน จาก Python

#### 2. ผลลัพธ์การเรียนรู้ (Learning Outcomes)

- ทักษะด้านวิทยาศาสตร์สิ่งแวดล้อม
- ทักษะการใช้โค้ด Python
- ทักษะการวิเคราะห์ข้อมูล

### 4. กลุ่มเป้าหมายและการบูรณาการกับการสอนวิทยาศาสตร์ (Target Learners and Integration with Science Teaching)

#### 1. ระดับชั้นหรือกลุ่มผู้เรียน

มุ่งสอนนักเรียนในระดับชั้นประถมศึกษาปีที่ 5-6

#### 2. หัวข้อทางวิทยาศาสตร์/สิ่งแวดล้อมที่ครอบคลุม

- โครงการนี้เชื่อมโยงกับหลักสูตรวิทยาศาสตร์ของนักเรียนชั้นประถมศึกษาปีที่ 6 ในเรื่องปรากฏการณ์เรือนกระจก และประเด็นที่เลือกมาใช้คือเรื่อง PM 2.5 และภาวะโลกร้อนซึ่งเป็นประเด็นปัญหาระดับโลก ที่พลเมืองโลกทุกคนควรตระหนักและร่วมกันหาแนวทางช่วยแก้ไขหรือลดปรากฏการณ์ดังกล่าว

#### 3. แนวทางทางวิชาการ/แนวปฏิบัติ

การเรียนรู้เกี่ยวกับปรากฏการณ์เรือนกระจกและ PM2.5 สามารถบูรณาการเข้ากับแนวทางการเรียนรู้แบบสืบเสาะหาความรู้ (Inquiry-Based Learning: IBL) ได้อย่างมีประสิทธิภาพ โดยใช้ Chatbot เป็นเครื่องมือสำคัญในการช่วยนักเรียนสืบค้นข้อมูล วิเคราะห์ และเชื่อมโยงแนวคิดทางวิทยาศาสตร์กับสถานการณ์จริง การจัดการเรียนรู้ในรูปแบบนี้จะช่วยให้นักเรียนสามารถสร้างองค์ความรู้ด้วยตนเองผ่านกระบวนการตั้งคำถาม ค้นหาคำตอบ และอภิปรายผลจากข้อมูลที่ได้รับ

## 5. การออกแบบโครงงานและอัลกอริทึม (Project Design and Algorithm)

### 1. ภาพรวมของโปรแกรม/เครื่องมือ

โปรแกรมจะเรียกใช้โมเดลภาษาจาก Google Gemini สำหรับตอบคำถามทั่วไป เป็นระบบที่ช่วยให้ผู้ใช้สามารถ สอบถามข้อมูลเกี่ยวกับคุณภาพอากาศ PM 2.5 และภาวะโลกร้อน ได้แบบเรียลไทม์ โดยใช้ โมเดลภาษาจาก Google Gemini ในการวิเคราะห์และตอบคำถาม นอกจากนี้ โปรแกรมยังดึงข้อมูลสภาพอากาศจริงจาก OpenWeather API เพื่อให้คำตอบที่อ้างอิงจากข้อมูลปัจจุบัน

### 2. อัลกอริทึมหรือผังงาน (Flowchart)

#### 1. การเริ่มต้น (Start)

- รับที่อยู่จากผู้ใช้
- แปลงที่อยู่เป็นพิกัด (Latitude, Longitude) โดยใช้ geopy

#### 2. ดึงข้อมูลจาก API

- เรียก OpenWeather API เพื่อดึงข้อมูล PM 2.5 และ สภาพอากาศ
- จัดรูปแบบข้อมูล JSON

#### 3. วิเคราะห์ข้อมูล

- แยกค่า PM 2.5, อุณหภูมิ, ความชื้น, และค่าฝุ่นอื่น ๆ
- ตรวจสอบระดับ AQI (Air Quality Index) และให้คำแนะนำ

#### 4.โต้ตอบกับผู้ใช้

- รับคำถามจากผู้ใช้เกี่ยวกับ PM 2.5 และภาวะโลกร้อน
- ส่งคำถามพร้อมข้อมูลปัจจุบันไปให้ Google Gemini AI
- รับคำตอบและแสดงผล

#### 5. แสดงผล

- แสดง ค่าฝุ่น PM 2.5, อุณหภูมิ และสภาพอากาศปัจจุบัน
- ให้คำแนะนำว่าควรทำกิจกรรมกลางแจ้งหรือไม่

#### 6. จบการทำงาน (End)

### 3. ฟังก์ชันสำคัญ (Key Functions)

- `get_air_pollution` ใช้สำหรับดึงข้อมูลคุณภาพอากาศของที่อยู่ โดยจะแปลงที่อยู่เป็นพิกัดแล้วใช้พิกัดนั้นเพื่อเรียก API

- `get_weather` ใช้สำหรับดึงข้อมูลสภาพอากาศของที่อยู่ โดยจะแปลงที่อยู่เป็นพิกัดแล้วใช้พิกัดนั้นเพื่อเรียก API
- `answer_question_gemini` ใช้สำหรับตอบคำถามทั้งหมดที่เข้ามา โดยมีการแนบเนื้อหาที่จำเป็นประกอบการตอบคำถาม เช่น สภาพอากาศ คุณภาพอากาศและเวลาปัจจุบัน

#### 4. ไลบรารีที่ใช้ (Libraries Used)

- `google-gemai` ใช้สำหรับเรียกใช้โมเดลภาษา Google Gemini สำหรับตอบคำถามทั่วไป
- `geopy` ใช้แปลงข้อมูลที่อยู่ที่เป็นตัวอักษรเป็นพิกัด latitude, longitude
- `requests` ใช้สำหรับเรียก API เพื่อดึงข้อมูลที่จำเป็น ในที่นี้จะดึงสภาพอากาศและคุณภาพอากาศ
- `datetime` ใช้เพื่อดึงข้อมูลเวลาปัจจุบันของแต่ละสถานที่

### 6. การพัฒนาโค้ด (Coding Implementation)

#### 1. โค้ด (Code Snippets)

เรียกใช้เครื่องมือสำหรับตอบคำถาม สามารถรองรับคำถามได้หลาย คล้าย ChatGPT

```
1. from google import genai
2.
3. # ใช้โมเดลภาษาชื่อ Gemini เป็นของ Google เป็นคนตอบคำถาม
4. GOOGLE_API_KEY="AIzaSyC_r6_tX6GXsedpXpxa5XGxMHsonemGPqg"
5. model_name = "gemini-2.0-flash"
6.
7. google_client = genai.Client(api_key=GOOGLE_API_KEY)
```

เรียกเครื่องมือเพื่อดูเวลาปัจจุบันที่ไทย

```
from datetime import datetime
import pytz

# เวลาปัจจุบัน
now = datetime.now(pytz.timezone('Asia/Bangkok'))

# เปลี่ยนรูปแบบของเวลาให้อยู่ในรูปแบบ Year-Month-Day Hour:Minute:Second
```

```
now = now.strftime("%Y-%m-%d %H:%M:%S")
```

### สร้างเครื่องมือที่แปลงที่อยู่เป็นพิกัด

```
import requests
from geopy.geocoders import Nominatim # เอาให้ latitude, longitude ของที่อยู่
import json

# สร้าง geocoder
geolocator = Nominatim(user_agent="Address_finder")

# รับชื่อเมือง -> แปลงเป็นพิกัด -> นำพิกัดไปหาค่าฝุ่น

def get_air_pollution(address):
    # แปลงที่อยู่เป็นพิกัด
    location = geolocator.geocode(address)
    if location: # ถ้าพบพิกัดของที่อยู่
        lat = location.latitude
        lon = location.longitude
        API_key = "50754ebf36fbb2d186e137ba47df18fa"
        url = f"http://api.openweathermap.org/data/2.5/air_pollution?lat={lat}&lon={lon}&appid={API_key}"
        response = requests.get(url)
        result = json.loads(response.text)["list"][0]
    else:
        result = "ไม่พบข้อมูลค่าฝุ่นของที่อยู่"
    return result

# รับชื่อเมือง -> แปลงเป็นพิกัด -> นำพิกัดไปหาค่าสภาพอากาศ

def get_weather(address):
    location = geolocator.geocode(address)
```

```

if location: # ถ้าพบพิกัดของที่อยู่
    lat = location.latitude
    lon = location.longitude
    API_key = "50754ebf36fbb2d186e137ba47df18fa"
    url =
f"https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&api
d={API_key}"
    response = requests.get(url)
    result = response.text
else:
    result = "ไม่พบข้อมูลสภาพอากาศของที่อยู่"
return result

```

```

# ฟังก์ชันสำหรับตอบคำถาม
def answer_question_gemini(question, now, air_quality_json, weather_json):
    # เป็นเหมือนคำสั่งที่จะเอาข้อมูล AI โดยเราจะนำข้อมูลที่เราหาได้แนบไปในเนื้อหา เช่น เวลาปัจจุบัน คำถาม สภาพอากาศ ค่าฝุ่น
    # จากนั้น AI จะใช้ข้อมูลทั้งหมดที่ให้ไป ตัดสินใจว่าควรตอบอะไรคำถามนั้น

    prompt = f"""
        Current datetime: {now}

        You are an AI assistant with expertise in environmental science,
        specializing in global warming, air pollution (particularly PM2.5), and
        weather conditions.

        Your role is to provide accurate and concise answers to user questions
        about these topics. Follow these guidelines:

        1. Respond to the user's question: "{question}".
        2. Keep answers short, simple, and scientifically correct (no more
        than 2-3 sentences).
        3. Use your built-in knowledge to address causes, effects, or
        solutions related to global warming, PM2.5, and weather.
    """

```

4. If the user provides external data in JSON format, use it to answer the question and reference it as "ข้อมูลของคุณ". Two types of JSON data may be provided:

- **\*\*Air Quality Data\*\*** (คุณภาพอากาศ):

```
{
  "dt": "Unix timestamp, UTC",
  "main": {{"aqi": "Air Quality Index (1=Good, 2=Fair,
3=Moderate, 4=Poor, 5=Very Poor)"}}},
  "components": {
    "co": "CO concentration, µg/m³",
    "no": "NO concentration, µg/m³",
    "no2": "NO2 concentration, µg/m³",
    "o3": "O3 concentration, µg/m³",
    "so2": "SO2 concentration, µg/m³",
    "pm2_5": "PM2.5 concentration, µg/m³",
    "pm10": "PM10 concentration, µg/m³",
    "nh3": "NH3 concentration, µg/m³"
  }
}
```

- **\*\*Weather Data\*\*** (สภาพอากาศ):

```
{
  "coord": {{"lon": "Longitude", "lat": "Latitude"}},
  "weather": [{"id": "Weather condition ID", "main": "Weather
group (e.g., Rain, Snow)", "description": "Weather description", "icon":
"Weather icon ID"}],
  "base": "Internal parameter",
  "main": {
    "temp": "Temperature (Metric: °C)",
    "feels_like": "Feels like temperature (Metric: °C)",
    "pressure": "Pressure, hPa",
    "humidity": "Humidity, %",
```



```

        "temp_min": "Min temperature (Metric: °C)",
        "temp_max": "Max temperature (Metric: °C)",
        "sea_level": "Sea level pressure, hPa",
        "grnd_level": "Ground level pressure, hPa"
    }},
    "visibility": "Visibility, meters",
    "wind": {"speed": "Wind speed, m/s", "deg": "Wind
direction, degrees", "gust": "Wind gust, m/s"}},
    "clouds": {"all": "Cloudiness, %"}},
    "rain": {"1h": "Rain volume, mm/h (if available)"},
    "snow": {"1h": "Snow volume, mm/h (if available)"},
    "dt": "Unix timestamp, UTC",
    "sys": {"country": "Country code", "sunrise": "Sunrise,
Unix UTC", "sunset": "Sunset, Unix UTC"}},
    "timezone": "Shift in seconds from UTC",
    "name": "City name"
}

```

5. If additional information from the web or X posts would improve the response, include it succinctly.

6. If the question is unclear, ask a short follow-up question for clarification.

7. Answer the question in THAI.

Your goal is to deliver quick, helpful insights. Below is the user's JSON data (if provided):

- Air Quality Data: {air\_quality\_json}
- Weather Data: {weather\_json}

Now, answer the user's question: "{question}".

""

```

# คำสั่งที่ส่งชุดคำถามไปถามเพื่อหาคำตอบ

response = google_client.models.generate_content(
    model=model_name,
    contents=[prompt])

return response.candidates[0].content.parts[0].text

from datetime import datetime

import pytz

# ให้ผู้ใช้ระบุที่อยู่มาก่อนตอนเริ่ม จะใช้ที่อยู่นี้แปลงเป็นพิกัดเอาไว้ดึงข้อมูลสภาพอากาศ ค่าฝุ่น

address = input("กรุณาระบุที่อยู่:")

while True:

    # เวลาปัจจุบัน

    now = datetime.now(pytz.timezone('Asia/Bangkok'))

    # เปลี่ยนรูปแบบของเวลาให้อยู่ในรูปแบบ Year-Month-Day Hour:Minute:Second

    now = now.strftime("%Y-%m-%d %H:%M:%S")

    # รับคำถามจากผู้ใช้

    question = input("\nสามารถถามคำถามเกี่ยวกับโลกร้อนหรือฝุ่น PM2.5 ได้เลย\nคำถาม:")

    # หาค่าฝุ่น สภาพอากาศ

    air_quality_json = get_air_pollution(address)

    weather_json = get_weather(address)

    answer = answer_question_gemini(question, now, air_quality_json,
    weather_json)

    print('\nคำตอบ: '+answer)

```

## คำอธิบายขั้นตอนหลัก

### 1. การจัดการข้อมูล (Data Handling)

โปรแกรมรับข้อมูล 3 ส่วนหลัก:

ข้อมูลวันเวลา – ใช้ datetime และ pytz ดึงเวลาปัจจุบันของไทย

ข้อมูลพิกัดที่อยู่ – ใช้ geopy แปลงชื่อเมืองเป็นพิกัดละติจูดและลองจิจูด

ข้อมูลสิ่งแวดล้อม (อากาศ & ฝุ่น PM 2.5) – ใช้ API ของ OpenWeather ดึง JSON

### 2. การใช้ Loops และ Conditionals

Loop: while True

- ใช้เพื่อให้ผู้ใช้ถามคำถามซ้ำได้เรื่อย ๆ
- ถ้าผู้ใช้พิมพ์ "exit" จะหยุดทำงาน

Conditionals: if-else

- ตรวจสอบว่า API ดึงข้อมูลสำเร็จหรือไม่ (if location:)
- ถ้าไม่พบข้อมูล จะส่ง "ไม่พบข้อมูลสภาพอากาศของที่นี่"

### 3. การใช้ฟังก์ชัน (Functions)

`get_air_pollution(address)`

1. แปลง address → latitude, longitude
2. เรียก API → ดึงค่าฝุ่น PM 2.5, PM10
3. คืนค่า JSON

`get_weather(address)`

1. แปลง address → latitude, longitude
2. เรียก API → ดึงอุณหภูมิ, ความชื้น, ลม, สภาพอากาศ
3. คืนค่า JSON

`answer_question_gemini(question, now, air_quality_json, weather_json)`

1. สร้าง Prompt สำหรับ AI
2. แนบข้อมูลฝุ่น, สภาพอากาศ → ให้ AI วิเคราะห์
3. คืนค่าคำตอบ

#### 4. การใช้ AI ตอบคำถาม

##### prompt ที่ส่งให้ AI

Current datetime: 2025-03-12 14:30:00

Your role: Environmental Science Expert

User's question: "ฝุ่น PM 2.5 วันนี้เป็นยังไง?"

User's data: {ข้อมูล PM2.5 และอากาศ}

##### การตอบกลับจาก AI

"ค่าฝุ่น PM2.5 ในพื้นที่ของคุณอยู่ที่ 23.4  $\mu\text{g}/\text{m}^3$  ซึ่งอยู่ในระดับปานกลาง ควรหลีกเลี่ยงกิจกรรมกลางแจ้งนานเกินไป"

#### การทดสอบและแก้บั๊ก (Testing and Debugging)

ปัญหาที่พบคือในครั้งแรกที่ไม่สามารถดึงข้อมูล API จากเว็บไซต์ตรวจสอบคุณภาพอากาศได้ และคำถามในการตอบ chatbot เมื่อนำไปใช้กับนักเรียนมีแนวคำตอบน้อยเกินไป ถ้าตอบไม่ตรงกับที่ระบุไว้ในโปรแกรม chatbot จะไม่สามารถทำงานได้ และวิธีการแก้ไขคือ ศึกษาวิธีการเชื่อมต่อ API เพิ่มจากแหล่งข้อมูล และปรึกษาผู้เชี่ยวชาญทำการพัฒนาชุดคำถามให้สามารถตอบได้ครอบคลุมมากขึ้นโดยดึงข้อมูลจาก AI Google Gemini เพื่อให้ตอบคำถามผู้เรียนได้ครอบคลุมมากขึ้น

ปัญหาอีกหนึ่งสิ่งที่พบคือ นิสิตพยายามเชื่อมต่อให้สร้างไลน์แชทบอทเพื่อแจ้งเตือนค่าฝุ่นกับไลน์ห้องเรียนแต่ยังดำเนินการไม่สำเร็จ เลยแก้ปัญหาโดยใช้ถามตอบ Q&A ใน python เลยแต่เพิ่มปริมาณคำถามให้ครอบคลุมกับคำตอบมากขึ้นโดยใช้ Gemini เข้ามาช่วย

#### 7. แผนการนำไปใช้ในชั้นเรียน (Classroom Implementation Plan)

##### 1. โครงสร้างบทเรียน (Lesson Outline)

การนำมาโปรแกรมมาใช้ในห้องเรียนจะนำโปรแกรมมาใช้โดยแบ่งเป็น 2 ชั้นเรียน ดังนี้

1. ชั้นเรียนชั้นประถมศึกษาปีที่ 5 ซึ่งนิสิตเป็นครูที่ปรึกษา ทางโรงเรียนได้มีมาตรการในการให้ทุกห้องเรียนตรวจเช็คคุณภาพอากาศทุกวันว่าสามารถทำกิจกรรมกลางแจ้งได้ตามปกติ หรือว่าควรหลีกเลี่ยง นิสิตจะนำแชทบอทที่พัฒนาขึ้นมาใช้ในการตรวจวัดคุณภาพอากาศโดยระบุพิกัดที่จังหวัดปราจีนบุรี ที่ตั้งของโรงเรียน นำแชทบอทแสดงขึ้นหน้าจอห้องเรียนเพื่อให้ผู้เรียนทราบ และสามารถซักถามแชทบอทได้ว่าตนเองสามารถออกไปเรียนในวิชากลางแจ้ง

2. ชั้นเรียนชั้นประถมศึกษาปีที่ 6 ใช้ในรายวิชาวิทยาศาสตร์พื้นฐาน เนื้อหาที่เรียนในหน่วยเรื่อง ปรากฏการณ์เรือนกระจก นักเรียนเรียนจะต้องสามารถอธิบายความหมายของปรากฏการณ์ เรือนกระจก ระบุว่าแก๊สเรือนกระจกประกอบไปด้วยอะไรบ้าง แต่ละแก๊สมีปริมาณสัดส่วนเท่าใด และสามารถระบุวิธีการป้องกันและลดการปล่อยแก๊สเรือนกระจก ดังนั้นนิสิตจะนำเครื่องมือ แห่บอห์นเป็นเครื่องมือให้นักเรียนในชั้นได้ใช้เป็นแหล่งการสืบค้นข้อมูล

- มีกิจกรรมและใบงานให้นักเรียน จำนวน 2 ชั่วโมง

## 2. ลำดับขั้นตอนการสอน (Instructional Sequence)

### ขั้นกระตุ้นความสนใจ (Engage)

1. เปิดรูปภาพแสดงผลกระทบของปรากฏการณ์เรือนกระจก เช่น ภาพน้ำแข็งขั้วโลกละลาย เมืองที่มี หมอกควันหนาแน่น หรือพื้นที่แห้งแล้ง
2. กระตุ้นการคิดด้วยคำถาม เช่น:
  - อุณหภูมิโลกเพิ่มขึ้นทุกปีเพราะอะไร
  - ภาวะโลกร้อนเกี่ยวข้องกับปรากฏการณ์เรือนกระจกหรือไม่ อย่างไร
3. ให้นักเรียนตั้งคำถามที่อยากรู้เกี่ยวกับภาวะโลกร้อนและ PM2.5

### ขั้นสำรวจและค้นหา (Explore)

1. นักเรียนใช้ Chatbot เพื่อค้นหาความหมายของก๊าซเรือนกระจกและผลกระทบต่อสภาพภูมิอากาศ
2. แบ่งกลุ่มนักเรียนออกเป็น 5 กลุ่มละความสามารถ โดยแต่ละกลุ่มจะได้รับบัตรภาพที่แตกต่างกัน ได้แก่:
  - **กลุ่ม 1:** ภาพการเผาป่า → ปล่อย CO<sub>2</sub> และ PM2.5
  - **กลุ่ม 2:** ภาพการปศุสัตว์ → ปล่อย CH<sub>4</sub> (มีเทน) จากการย่อยอาหารของสัตว์เคี้ยวเอื้อง
  - **กลุ่ม 3:** ภาพโรงงานอุตสาหกรรม → ปล่อย CO<sub>2</sub>, NO<sub>2</sub> และ PM2.5 จากการเผาไหม้เชื้อเพลิง ฟอสซิล

- **กลุ่ม 4:** ภาพการใส่ปุ๋ยไนโตรเจน → ปปล่อย  $N_2O$  (ไนตรัสออกไซด์) จากกระบวนการย่อยสลายปุ๋ย
  - **กลุ่ม 5:** ภาพเครื่องปรับอากาศ → ปปล่อยสาร CFCs และ HFCs ซึ่งเป็นก๊าซเรือนกระจก
3. แต่ละกลุ่มต้องช่วยกันสืบค้นว่าภาพที่ได้รับเกี่ยวข้องกับก๊าซเรือนกระจกอย่างไร และปริมาณสัดส่วนของก๊าซเหล่านี้ในชั้นบรรยากาศเป็นเท่าใด
4. นักเรียนค้นหาข้อมูลเพิ่มเติมจากหนังสือเรียน และสามารถสอบถามเพิ่มเติมจาก Chatbot หรือครู

### ชั้นอธิบายและลงข้อสรุป (Explain)

1. ตัวแทนนักเรียนของแต่ละกลุ่มนำเสนอผลการสืบค้นหน้าชั้นเรียนเกี่ยวกับ:
- ความหมายของปรากฏการณ์เรือนกระจก
  - ก๊าซเรือนกระจกแต่ละชนิดและสัดส่วนในชั้นบรรยากาศ
  - แหล่งกำเนิดของก๊าซเรือนกระจกจากกิจกรรมของมนุษย์
  - การเชื่อมโยงระหว่างกิจกรรมที่กลุ่มได้รับกับปรากฏการณ์เรือนกระจก
2. ครูช่วยอธิบายเสริมและให้ข้อสังเกตเกี่ยวกับข้อมูลที่นำเสนอ

### ชั้นขยายความรู้ (Elaborate)

1. นำเสนอสถานการณ์ใหม่เกี่ยวกับข่าวปัญหาฝุ่น PM2.5 และผลกระทบต่อสุขภาพและสิ่งแวดล้อม
2. ให้นักเรียนช่วยกันค้นหาข้อมูลและอภิปรายว่า:
- การกระทำของมนุษย์ที่ก่อให้เกิดก๊าซเรือนกระจก มีผลกระทบหรือเกี่ยวข้องกับปริมาณฝุ่น PM2.5 หรือไม่ อย่างไร
  - PM2.5 ส่งผลต่อสภาพภูมิอากาศโดยรวมอย่างไร
  - เราจะลดการปล่อย PM2.5 และก๊าซเรือนกระจกได้อย่างไรบ้าง

### ชั้นประเมิน (Evaluate)

1. นักเรียนทำ **อนุทินสะท้อนการเรียนรู้** โดยเขียนตอบใน 3 ประเด็นต่อไปนี้:
  - สิ่งที่ได้เรียนรู้ในวันนี้
  - สิ่งที่ยังสงสัย
  - ความรู้สึกของการเรียนในครั้งนี้
2. ครูอ่านและสะท้อนความคิดเห็นของนักเรียน พร้อมให้ข้อเสนอแนะเพิ่มเติม

## 8. ผลลัพธ์และข้อสังเกต (Results and Observations)

### 1. ความสนใจของนักเรียน (Student Engagement)

เมื่อนำ Chatbot ไปใช้ในห้องเรียน นักเรียนให้ความสนใจอย่างมาก โดยมีการโต้ตอบกับ Chatbot อย่างต่อเนื่อง ทั้งในเรื่องที่เกี่ยวข้องกับ PM 2.5 และภาวะโลกร้อน รวมถึงการตั้งคำถาม นอกเหนือจากบทเรียน เช่น การเชื่อมโยงมลพิษทางอากาศกับสุขภาพ หรือผลกระทบต่อสิ่งแวดล้อมในระยะยาว สะท้อนให้เห็นถึง ความกระตือรือร้นในการเรียนรู้ และ การคิดวิเคราะห์ที่ลึกซึ้ง อย่างไรก็ตาม นักเรียนบางคนอาจพบอุปสรรคในช่วงแรก โดยเฉพาะการทำความเข้าใจการใช้ Chatbot แต่เมื่อนักเรียนได้ฝึกใช้ไปสักระยะก็สามารถใช้งานได้อย่างคล่องแคล่วมากขึ้น นอกจากนี้ การมี Chatbot ตอบคำถามแบบโต้ตอบยังช่วยให้นักเรียนรู้สึกสนุกกับการเรียน และกระตุ้นให้นักเรียนอยากรวบรวมข้อมูลสิ่งแวดล้อมเพิ่มเติม

### 2. พัฒนาการด้านการเรียน (Learning Gains)

จากการตรวจใบงานและกิจกรรมของนักเรียน พบว่านักเรียนสามารถ ดึงข้อมูลจาก Chatbot มาใช้ในการตอบคำถามและวิเคราะห์ข้อมูลได้อย่างถูกต้อง ซึ่งแสดงให้เห็นว่านักเรียนสามารถ เชื่อมโยงข้อมูลจากแหล่งดิจิทัลเข้ากับบทเรียนทางวิทยาศาสตร์ได้ นอกจากนี้ ในระหว่างการใช้ Chatbot ครูสังเกตเห็นว่านักเรียนมีความกระตือรือร้นในการสำรวจข้อมูล โดยพยายามตั้งคำถามที่หลากหลาย พยายามใส่คำถามนอกเหนือบทเรียนหรือไม่เกี่ยวข้องกับบทเรียนเลยเพื่อทดสอบความสามารถในการตอบคำถามของ chatbot

ด้านทักษะโค้ด นักเรียนที่สนใจสามารถเรียนรู้หลักการ API, JSON และการใช้ Python เพื่อดึงข้อมูลสภาพอากาศ ผ่านโค้ดที่ครูนำเสนอ ซึ่งช่วยให้พวกเขาเข้าใจแนวคิดเกี่ยวกับ การเขียน โปรแกรมเพื่อแก้ปัญหาทางสิ่งแวดล้อม นอกจากนี้ นักเรียนที่มีความสนใจด้านดิจิทัลสามารถ ทดลอง

ปรับแต่งคำถาม หรือพยายามหาวิธีเชื่อมต่อ Chatbot กับข้อมูลอื่น ๆ ซึ่งเป็นการกระตุ้นให้นักเรียนสนใจการใช้ได้มากขึ้น

### 3. อุปสรรคหรือปัญหา (Challenges)

1. ข้อจำกัดในเรื่องอุปกรณ์คอมพิวเตอร์ ทางโรงเรียนมีจำนวนคอมพิวเตอร์ไม่เพียงพอต่อนักเรียนต้องใช้ 2 คนต่อหนึ่งเครื่อง
2. นักเรียนไม่สามารถใช้โทรศัพท์ในการปฏิบัติกิจกรรมได้เนื่องจากทางโรงเรียนไม่อนุญาตให้นำโทรศัพท์มาโรงเรียน การสื่อสารกับแชทบอทจึงมีช่องทางน้อย

## 9. การสะท้อนและอภิปราย (Reflection and Discussion)

### 1. สะท้อนตนเอง (Self-Reflection)

- ส่วนไหนของโครงการที่ดี หรือส่วนไหนที่ควรปรับปรุง?

### 2. ข้อเสนอแนะจากเพื่อน (Peer Feedback)

- สรุปความคิดเห็นหรือคำแนะนำที่ได้จาก “show and share” ในชั้นเรียน
- จะนำไปปรับโค้ดหรือปรับกิจกรรมการสอนอย่างไร?

### 3. ผลกระทบต่อการสอน (Implications for Teaching)

- แนวทางการนำโค้ดผสมกับการสอนครั้งต่อไปหรือในวิชาอื่น
- การต่อยอดหรือขยายผลในอนาคต

## 10. สรุปผลและแนวทางในอนาคต (Conclusion and Future Directions)

โครงการ "AI Chatbot for PM 2.5 and Climate Awareness" มีเป้าหมายในการพัฒนา Chatbot อัจฉริยะที่สามารถให้ข้อมูลเกี่ยวกับฝุ่น PM 2.5 และภาวะโลกร้อน โดยใช้ Python, Google Gemini AI และ OpenWeather API เพื่อช่วยให้นักเรียนสามารถสอบถามข้อมูลคุณภาพอากาศแบบเรียลไทม์ โครงการนี้มุ่งเน้นการบูรณาการเทคโนโลยีดิจิทัลเข้ากับการเรียนการสอน ส่งเสริมให้นักเรียนฝึกทักษะการคิดวิเคราะห์และการใช้เครื่องมือดิจิทัลเพื่อประเมินสถานการณ์มลพิษทางอากาศ

Chatbot ที่พัฒนาขึ้นสามารถ ดึงข้อมูลค่าฝุ่น PM 2.5 และสภาพอากาศปัจจุบันจาก API และให้คำตอบโดยอิงข้อมูลจริง ทำให้ผู้เรียนสามารถใช้เป็นเครื่องมือช่วยตัดสินใจ เช่น ควรทำกิจกรรมกลางแจ้งหรือควรป้องกัน



ตัวเองจากมลพิษ นักเรียนในระดับ ประถมศึกษาปีที่ 5 สามารถใช้ตรวจสอบคุณภาพอากาศประจำวัน ขณะที่ นักเรียน ประถมศึกษาปีที่ 6 สามารถใช้ Chatbot เป็นแหล่งข้อมูลในการศึกษาเรื่องภาวะโลกร้อนและผลกระทบของก๊าซเรือนกระจก

ผลงานที่ได้จากโครงการนี้ ไม่เพียงแต่ช่วยให้นักเรียนเข้าใจ แหล่งกำเนิดและผลกระทบของ PM 2.5 แต่ยังเสริมสร้างทักษะ การใช้โค้ด Python, การอ่านค่าจาก API และการใช้ AI เพื่อการวิเคราะห์ข้อมูล นอกจากนี้ยังสามารถต่อยอดพัฒนาให้ แจ้งเตือนค่าฝุ่นผ่าน LINE Notify, ใช้ Machine Learning ทำนายค่าฝุ่นในอนาคต และเปรียบเทียบคุณภาพอากาศของแต่ละพื้นที่ เพื่อสร้างองค์ความรู้ที่ลึกซึ้งยิ่งขึ้นเกี่ยวกับมลพิษทางอากาศ โครงการนี้จึงเป็นการบูรณาการเทคโนโลยีเข้ากับการเรียนรู้ด้านสิ่งแวดล้อม ช่วยให้นักเรียนตระหนักถึงปัญหามลพิษทางอากาศ พร้อมทั้งพัฒนาทักษะการวิเคราะห์ข้อมูลและการใช้เครื่องมือดิจิทัล ซึ่งเป็นทักษะสำคัญในศตวรรษที่ 21

## 11. บรรณานุกรม (References)

Google. (n.d.). *Google Generative AI for Python (Gemini API)*. Google AI. Retrieved March 12, 2025, from <https://ai.google.dev/>

Python Software Foundation. (n.d.). *datetime — Basic date and time types*. Python 3. Retrieved March 12, 2025, from <https://docs.python.org/3/library/datetime.html>

Stubbs, S. (n.d.). *pytz: World Timezone Definitions for Python*. Retrieved March 12, 2025, from <https://pypi.org/project/pytz/>

Reitz, K., & Team. (n.d.). *Requests: HTTP for Humans*. Retrieved March 12, 2025, from <https://docs.python-requests.org/en/latest/>

Denisov, K. (n.d.). *Geopy: Geocoding library for Python*. Retrieved March 12, 2025, from <https://geopy.readthedocs.io/en/stable/>

OpenWeather. (n.d.). *Current and forecast weather data API*. Retrieved March 12, 2025, from <https://openweathermap.org/api>