# IBM NaanMuthalvan – Artificial Inteligence Group 4

College name: SSM Institute Of Engineering And Technology.

College code: 9221

#### **Team Members:**

J.Margrate Sneka (922121106045)

S.Nagapriya (922121106056)

M.Pooja (922121106064)

C.M.Priyadharshini (922121106069)

#### Phase 3:

MEASURE ENERGY CONSUMPTION

#### **Process:**

#### Step1:

Gather the necessary data

#### Step2:

This typically involves collecting energy consumption data from various sources

### Step3:

Clean and preprocess the data to remove outliers, handle missing values, and format it for analysis

### Step4:

Create relevant features from the data, such as time of day, weather conditions, or occupancy patterns, which could impact energy consumption

### Step5:

Choose the appropriate AI techniques for our project

### Step6:

Train our AI models using the preprocessed data

### Step7:

Implement anomaly detection algorithms to identify irregular or inefficient energy consumption patterns.

### Step8:

Develop visualization tools to present the results

### Step9:

Consider how our project can be scaled to accommodate more extensive energy monitoring, if needed

# **Python Coding:**

import random

import time

from datetime import datetime

# Simulated energy consumption data generator def generate\_energy\_data():

current\_time = datetime.now()

while True:

timestamp = current\_time.strftime("%Y-%m-%d %H:%M:%S")

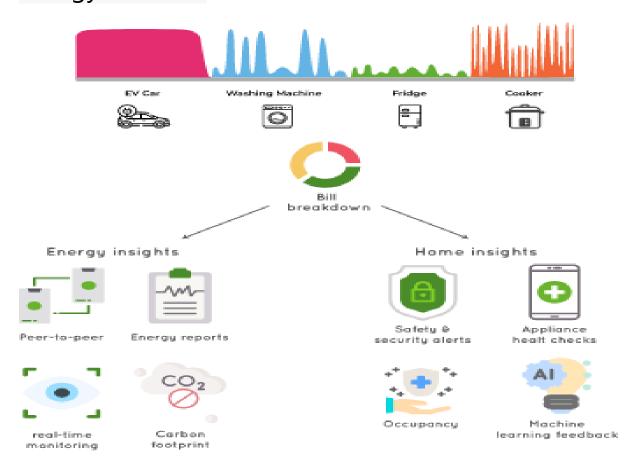
energy\_consumption = random.uniform(100, 1000) # Simulated consumption in watts

yield timestamp, energy\_consumption

time.sleep(1) # Simulate data update every second

```
# Data storage (you can use a database or file storage in a real
system)
def store_energy_data(timestamp, energy_consumption):
  with open('energy_data.csv', 'a') as f:
    f.write(f'{timestamp},{energy_consumption}\n')
if __name__ == "__main__":
  data_generator = generate_energy_data()
  try:
    while True:
         timestamp, energy_consumption = next(data_generator)
       store_energy_data(timestamp, energy_consumption)
       print(f"Timestamp: {timestamp}, Energy Consumption:
{energy_consumption} W")
  except KeyboardInterrupt:
    print("Monitoring stopped.")
```

# **Energy monitor:**



### Solution:

- Smart meters with built-in AI algorithms can continuously collect and analyze energy consumption data at a granular level, often in real-time
- Machine learning models can identify patterns, anomalies, and trends in energy consumption

- Al can forecast future energy consumption based on historical data and external factors like weather forecasts
- Some devices can be integrated with smart home or building systems to remotely control appliances, thermostats, lighting, and other devices to optimize energy usage
- Many of these devices have companion apps that provide users with easy access to energy consumption data and insights on their mobile devices
- This information can motivate users to reduce their environmental impact

#### Conclusion:

Incorporating AI into energy consumption management provides a pathway to reducing energy costs, minimizing environmental impact, and promoting sustainability. It empowers individuals, businesses, and utilities to make data-driven decisions, ultimately contributing to more efficient and responsible energy use. Regular updates and maintenance are crucial to keep AI algorithms accurate and relevant.