VELAMMAL INSTITUTE OF TECHNOLOGY

Approved by AICTE - New Delhi
Affiliated to Anna University - Chennai
Accredited by NBA & NAAC

NAAN MUDHALVAN PROJECT(IBM)

IBM AI 101 ARTIFICIAL INTELLIGENCE-GROUP 1

Title : Measure Energy Consumption

Team name: Proj_224826_Team_1

Phase : Phase 3

Team members: SIVASUBRAMANI C (reg no 1133211036090)

THIRUMURUGAN S(reg no 113321106107)

SANTHOSH R(reg no 113321106083)

SRINIVASA PRABHU(reg no 113321106095)

SAISHANKAR(reg 113321106302)

Problem Statement:

The problem at hand is to create an automated system that measures energy consumption, analyzes the data, and provides visualizations for informed decision-making. This solution aims to enhance efficiency, accuracy, and ease of understanding in managing energy consumption across various sectors.

Targeted area: Home power consumption.

Project Description: This project aims to develop an end-to-end solution for collecting energy consumption data, performing real-

time analysis, and providing insightful visualizations to support informed decision-making. The system will enhance efficiency, accuracy, and ease of understanding in managing energy consumption across various sectors.

Project Components:

1. **Data Collection and Integration**:
    - **Data Sources**:
        - Smart Meters
        - IoT Sensors
        - Building Management Systems (BMS)
        - Weather Data
        - Energy Management Systems (EMS)
    - **Data Collection**:
        - Implement data collection modules to retrieve data from various sources.
        - Integrate data collection processes to ensure data from different sources is consolidated.

Four main steps to be follow

--> Data Sources

--> Data Collection

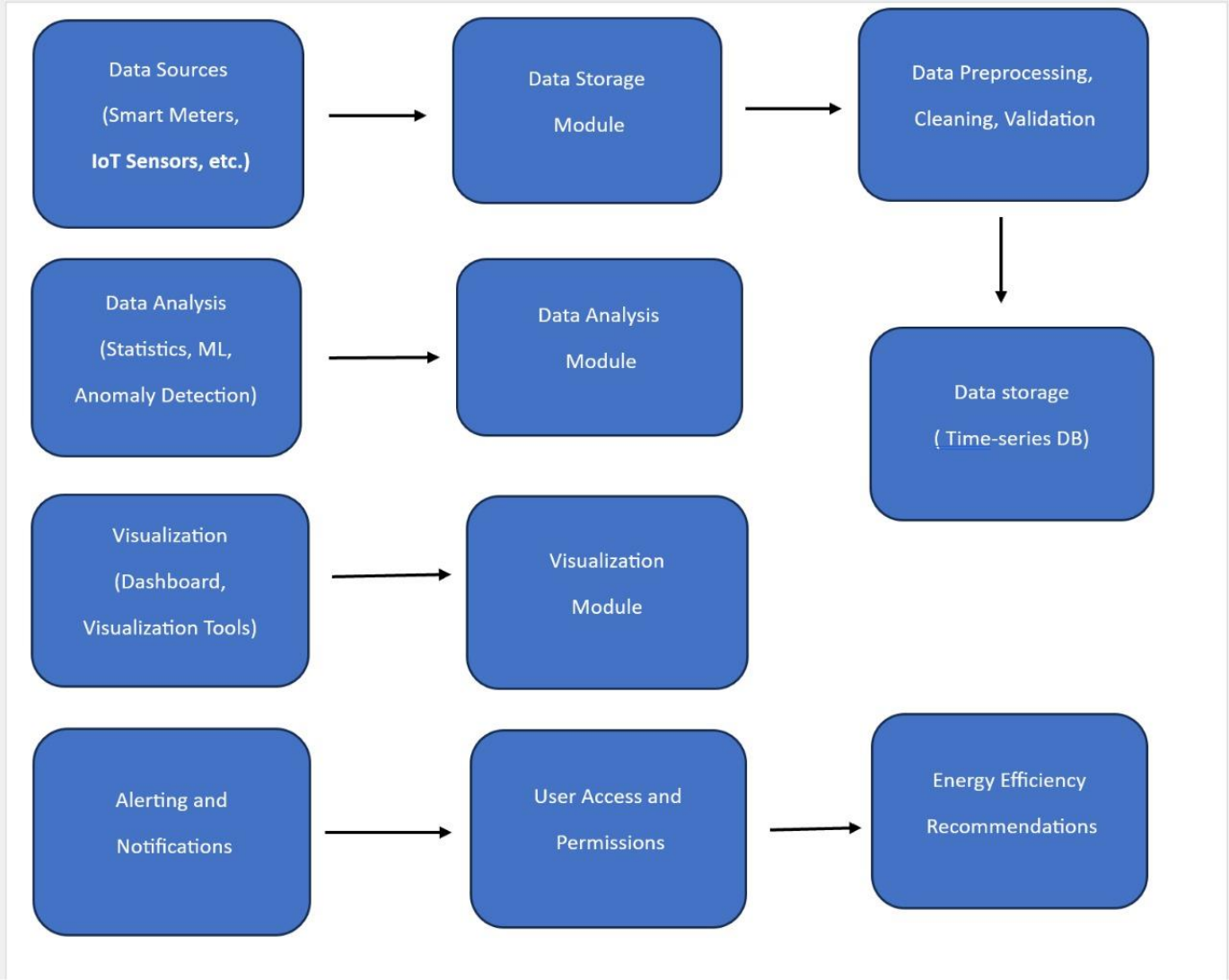
--> Data Storage

--> Data Preprocessing

Block Diagram:

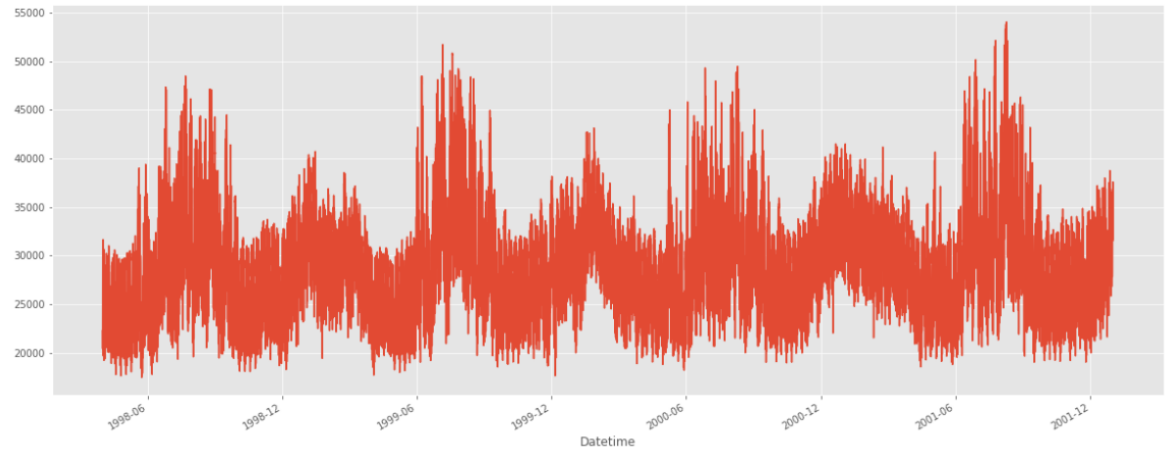
A block diagram represents the major components and their relationships in the system. In this case, the block diagram would

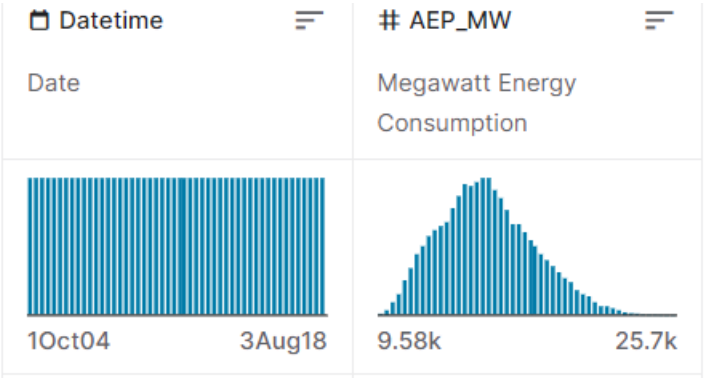
depict the flow of data and operations within the system. Here's a textual representation of the block diagram:



Data set Analysis Techniques:

Use techniques such as regression analysis, time series analysis, clustering, or machine learning to extract valuable insights from the data. For your energy consumption data, you might want to explore seasonality, trends, and potential anomalies.





The above image was collected from koggel dataset

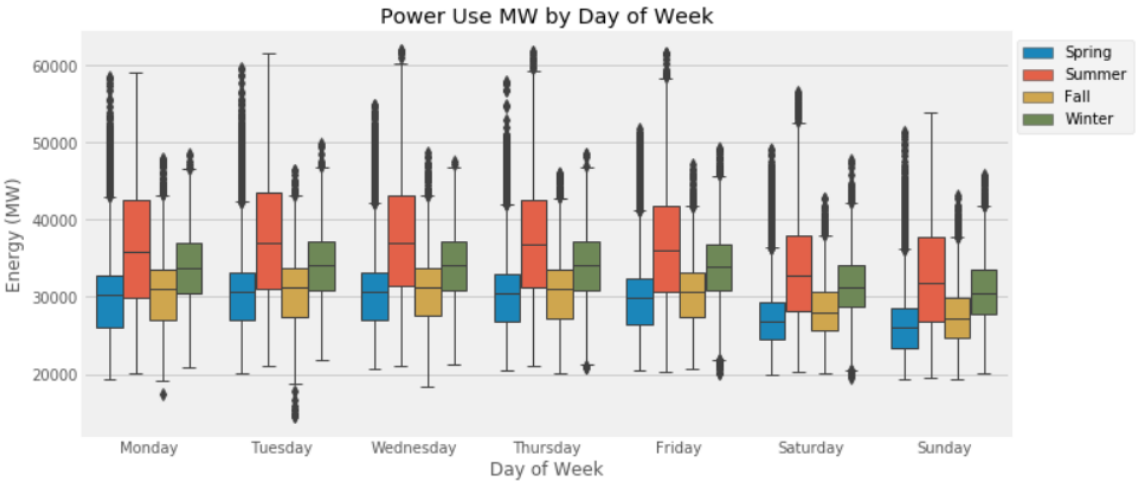
Program

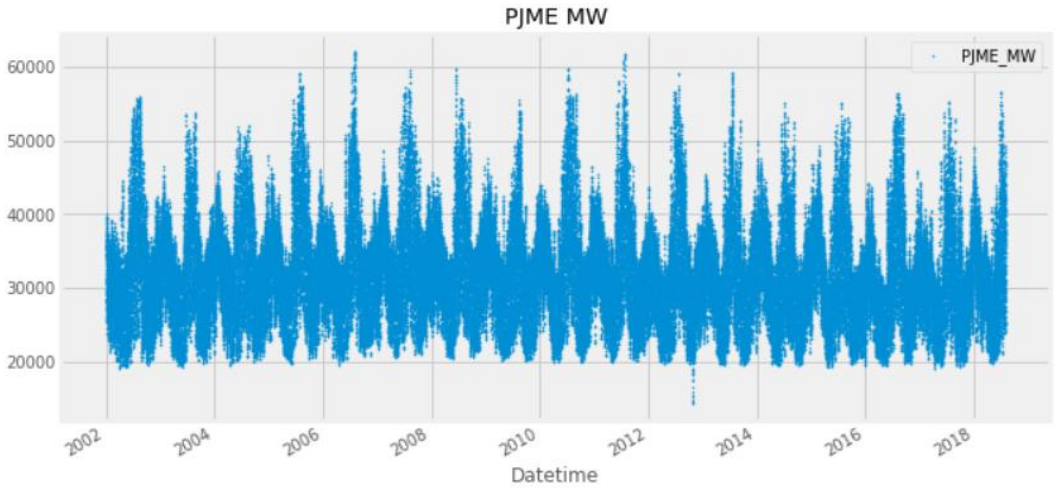
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from fbprophet import Prophet

from sklearn.metrics import mean_squared_error, mean_absolute_error
```

```
import warnings
warnings.filterwarnings("ignore")

plt.style.use('ggplot')
plt.style.use('fivethirtyeight')

def mean_absolute_percentage_error(y_true, y_pred):
    """Calculates MAPE given y_true and y_pred"""
    y_true, y_pred = np.array(y_true), np.array(y_pred)
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

pjme = pd.read_csv('../input/PJME_hourly.csv',
                   index_col=[0],
                  parse_dates=[0])
pjme.head()
Time Series Features
from pandas.api.types import CategoricalDtype

cat_type = CategoricalDtype(categories=['Monday','Tuesday',
                                        'Wednesday',
                                        'Thursday','Friday',
                                        'Saturday','Sunday'],

                            ordered=True)

def create_features(df, label=None):
    """
    Creates time series features from datetime index.
    """
```

```
    df = df.copy()
    df['date'] = df.index
    df['hour'] = df['date'].dt.hour
    df['dayofweek'] = df['date'].dt.dayofweek
    df['weekday'] = df['date'].dt.day_name()
    df['weekday'] = df['weekday'].astype(cat_type)
    df['quarter'] = df['date'].dt.quarter
    df['month'] = df['date'].dt.month
    df['year'] = df['date'].dt.year
    df['dayofyear'] = df['date'].dt.dayofyear
    df['dayofmonth'] = df['date'].dt.day
    df['weekofyear'] = df['date'].dt.weekofyear
    df['date_offset'] = (df.date.dt.month*100 + df.date.dt.day - 320)%1300

    df['season'] = pd.cut(df['date_offset'], [0, 300, 602, 900, 1300],
                          labels=['Spring', 'Summer', 'Fall', 'Winter']
                    )
    X = df[['hour','dayofweek','quarter','month','year',
           'dayofyear','dayofmonth','weekofyear','weekday',
           'season']]
    if label:
       y = df[label]
       return X, y
    return X

X, y = create_features(pjme, label='PJME_MW')
features_and_target = pd.concat([X, y], axis=1)
fig, ax = plt.subplots(figsize=(10, 5))
sns.boxplot(data=features_and_target.dropna(),
            x='weekday',
```

```
        y='PJME_MW',
        hue='season',
        ax=ax,
        linewidth=1)
ax.set_title('Power Use MW by Day of Week')
ax.set_xlabel('Day of Week')
ax.set_ylabel('Energy (MW)')
ax.legend(bbox_to_anchor=(1, 1))
plt.show()
```





conclusion

The automated energy consumption management system efficiently collects and analyzes data from various sources, offering insights for informed decision-making. Interactive visualizations enhance data exploration, while robust security and compliance measures protect sensitive information. Scalability and integration ensure adaptability, and ongoing support and improvement sustain system reliability, making it a valuable asset for optimizing energy usage and promoting sustainability.