A Project Report on

Energy Saver

Submitted in partial fulfilment of requirements for the award of the course of

**ADI1201 – DATA EXPLORATION AND VISUALIZATION**

Under the guidance of

**Mr.R. GOKULNATH MCA.,**

**IBM Software Technical Trainer/AI&DS**

Submitted By

**DHARSHINI K GAYATHRI C JEEVA K**

**(927623BAD023) (927623BAD034) (927623BAD045)**

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE**

# M.KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous)

**KARUR – 639 113**

DECEMBER 2024

**M. KUMARASAMY COLLEGE OF ENGINEERING**

**(Autonomous Institution affiliated to Anna University, Chennai)**

**KARUR – 639 113**

**BONAFIDE CERTIFICATE**

Certified that this project report on **“Energy Saver”** is the bonafide work of **DHARSHINI.K (927623BAD023), GAYATHRI C (927623BAD034) ,JEEVA K (927623BAD045)** who carried out the project work during the academic year 2023- 2024 under my supervision.



|  |  |
| --- | --- |
| Signature | Signature |
| **Mr.R.GOKULNATH MCA** | **Dr. A.SELVI Ph.D.,** |
| **IBM Software Technical Trainer** | **HEAD OF THE DEPARTMENT,** |
| Department of Artificial Intelligence, | Department of Artificial Intelligence, |
| M. Kumarasamy College of Engineering, | M. Kumarasamy College of Engineering, |
| Thalavapalayam, Karur -639 113. | Thalavapalayam, Karur -639 113. |

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**VISION OF THE INSTITUTION**

To emerge as a leader among the top institutions in the field of technical education

**MISSION OF THE INSTITUTION**

* Produce smart technocrats with empirical knowledge who can surmount the global challenges
* Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students
* Maintain mutually beneficial partnerships with our alumni, industry, and Professional associations

**VISION OF THE DEPARTMENT**

To achieve education and research excellence in Computer Science and Engineering

**MISSION OF THE DEPARTMENT**

* To excel in academic through effective teaching learning techniques
* To promote research in the area of computer science and engineering with the focus on innovation
* To transform students into technically competent professionals with societal and ethical

responsibilities

**PROGRAM EDUCATIONAL OBJECTIVES (PEOs)**

**PEO 1:** Graduates will have successful career in software industries and R&D divisions through continuous learning.

**PEO 2:** Graduates will provide effective solutions for real world problems in the key domain of computer science and engineering and engage in lifelong learning.

**PEO 3:** Graduates will excel in their profession by being ethically and socially responsible.

**PROGRAM OUTCOMES (POs)**

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in

diverse teams, and in multidisciplinary settings.

1. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
2. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
3. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**PROGRAM SPECIFIC OUTCOMES (PSOs)**

1. **PSO1: Professional Skills:** Ability to apply the knowledge of computing techniques to design and develop computerized solutions for the problems.
2. **PSO2: Successful career:** Ability to utilize the computing skills and ethical values in creating a successful career.

# ABSTRACT

This project centers on a comprehensive exploration of energy consumption data collected from smart meters, leveraging Python's pandas library for advanced data manipulation and analysis. With the growing emphasis on sustainability and the need for efficient energy usage, this study aims to provide meaningful insights into consumption behaviors. By employing a combination of statistical methods and data visualization techniques, it seeks to uncover hidden patterns, detect inefficiencies, and empower users with actionable knowledge to optimize their energy usage.

The analysis process begins with data cleaning and preprocessing to ensure the dataset is accurate, consistent, and ready for exploration. Key statistical analyses will be performed to identify correlations, anomalies, peak consumption periods, and seasonal trends, offering a nuanced understanding of how energy is utilized. The study will also utilize cutting-edge visualization tools to create intuitive and compelling graphical representations, such as time- series plots, distribution histograms, and heatmaps. These visuals will transform complex datasets into accessible and engaging narratives, enabling users to comprehend their energy usage effectively.

# ABSTRACT WITH POs AND PSOs MAPPING

|  |  |  |
| --- | --- | --- |
| **ABSTRACT** | **POs**  **MAPPED** | **PSOs**  **MAPPED** |
|  | **PO1(2)** | **PSO1(3)** |
| This project centers on a comprehensive exploration of | **PO2(3)**  **PO3(2)**  **PO4(2)**  **PO5(3)**  **PO6(1)**  **PO7(3)**  **PO8(2)**  **PO9(3) PO10(3) PO11(2) PO12(2)** | **PSO2(2)** |
| energy consumption data collected from smart meters, |  |
| leveraging Python's pandas library for advanced data |  |
| manipulation and analysis. With the growing emphasis on |  |
| sustainability and the need for efficient energy usage, this |  |
| study aims to provide meaningful insights into consumption |  |
| behaviors. By employing a combination of statistical |  |
| methods and data visualization techniques, it seeks to |  |
| uncover hidden patterns, detect inefficiencies, and empower |  |
| users with actionable knowledge to optimize their energy |  |
| usage. |  |
| Note: 1- Low, 2-Medium, 3- High |  |

**SUPERVISOR HEAD OF THE DEPARTMENT**

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER**  **No.** | **TITLE** | **PAGE**  **No.** |
|  | **ABSTRACT** | vi |
| **1** | **INTRODUCTION** | 1 |
|  | 1.1 Objective | 1 |
|  | 1.2 Overview | 1 |
|  | 1.3 Java Programming concepts | 2 |
| **2** | **PROJECT METHODOLOGY** | 4 |
|  | 2.1Agile methodology | 4 |
|  | 2.2 Block Diagram | 6 |
| **3** | **MODULE DESCRIPTION** | 7 |
|  | 3.1 Data collection and preprocessing | 7 |
|  | 3.2 Statistical analysis | 7 |
|  | 3.3 Time series analysis | 7 |
|  | 3.4 Visualization | 7 |
|  | 3.5 Energy efficiency recommendations | 7 |
| **4** | **RESULTS AND DISCUSSION** | 8 |
| **5** | **CONCLUSION** | 9 |
|  | **REFERENCES** | 9 |
|  | **APPENDIX** | 10 |
|  |  |  |

**CHAPTER 1**

# INTRODUCTION

With the growing focus on sustainability and energy conservation, smart meters have become essential tools for monitoring energy consumption. These devices provide real-time data, helping both consumers and utility providers make informed decisions. However, the vast amount of data generated requires effective analysis to extract valuable insights. This project utilizes Python's pandas library to analyze smart meter data, uncovering consumption patterns and trends. By identifying anomalies and offering recommendations for energy-saving measures, we aim to help users reduce costs and minimize their environmental impact, contributing to more sustainable energy practices.

## 1.1 Objective

The primary objective of the "Energy Saver: Analyzing Smart Meter Data for Efficiency" project is to develop a Java-based solution that processes and analyzes smart meter energy consumption data. The project aims to identify usage patterns, detect anomalies, and provide actionable insights to optimize energy usage, reduce costs, and promote sustainability.

**1.2 Overview**

This project focuses on analyzing energy consumption data collected from smart meters to identify patterns, detect anomalies, and provide actionable recommendations for reducing energy usage and costs. Utilizing Python's pandas library for data manipulation and visualization tools like Matplotlib, Seaborn, and Plotly, the project processes historical energy data to extract insights, optimize energy efficiency, and promote sustainability. The application incorporates key features such as time series analysis, statistical summaries, and dynamic visualizations, while also providing a user-friendly interface to upload and analyze dataset

## 1.3Concepts Used:

The "Energy Saver" project leverages several key Java programming concepts to effectively analyze smart meter data and provide actionable insights. File Handling plays a crucial role, as the application reads and writes large datasets in formats such as CSV or JSON, ensuring smooth data input and output. Data Structures like ArrayList, HashMap, and TreeMap are employed to store and manage energy consumption data efficiently, facilitating quick access and manipulation of records.

## Data Manipulation:

Using pandas for cleaning, transforming, and aggregating data.It refers to the process of organizing, restructuring, or transforming data to make it more suitable for analysis or visualization. It involves cleaning, formatting, and reshaping datasets to extract insights. In R, powerful packages like dplyr and reshape2 simplify these tasks.

## Data Visualization:

Leveraging libraries like Matplotlib and Seaborn to create insightful visualizations.It is the graphical representation of data and information. By using visual elements like charts, graphs, and maps, data visualization makes complex datasets easier to understand and analyze. It is an essential step in exploratory data analysis and communicating insights effectively.

## Time Series Analysis:

Utilizing pandas and statsmodels for trend and seasonal decomposition.It involves examining data points collected or recorded over time to identify trends, patterns, and seasonal variations, and to make forecasts or informed decisions. It is widely used in fields such as finance, economics, climate science, and more.

## Descriptive Statistics:

Calculating measures of central tendency and dispersion.It provide a summary of a dataset, highlighting key features and central tendencies, variability, and the overall distribution of the data. They help in understanding the basic characteristics of the data before diving into further analysis.

## Data Aggregation:

Grouping and summarizing data to extract meaningful patterns (e.g., daily/weekly/monthly aggregates).it involves combining data from multiple records or observations into summary forms to facilitate analysis. This technique is often used to analyze large datasets, identify patterns, and support decision-making processes.

## Outlier Detection:

Identifying and handling anomalies using statistical methods.Integration with External APIs (Optional): Fetching weather or appliance usage data to enhance analysis.It is the process of identifying data points that deviate significantly from the rest of the dataset. Outliers can indicate variability in measurements, experimental errors, or novel phenomena. Identifying outliers is crucial for cleaning data, ensuring accurate analysis, and avoiding misleading results.

## Optimization:

Suggesting actionable measures based on the data analysis.It is the process of finding the best solution to a problem from a set of possible solutions, often under specific constraints. It aims to maximize or minimize an objective function, such as profit, cost, time, or resource use.

# CHAPTER 2

**PROJECT METHODOLOGY**

## Agile Methodology

The Agile approach ensures flexibility, iterative development, and continuous improvement throughout the project life cycle. The process is divided into six categories:

## Data Collection

* + **Objective:** Gather fuel levels, location, and vehicle health metrics using sensors

and GPS modules.

* + **Process:** Sensors collect real-time data, which is processed by the Raspberry Pi Pico’s ADC.

## Data Transmission

* + **Objective:** Transmit collected data to a central system or user device.
  + **Process:** Use wireless communication (e.g., Bluetooth, Wi-Fi, or GSM) for

seamless and efficient data transfer.

## Data Storage

* + **Objective:** Log data for analysis and historical record-keeping.
  + **Process:** Store timestamps and sensor readings locally (on an SD card or the Raspberry Pi Pico) or on a cloud server.

## Data Analysis

* + **Objective:** Process and interpret the collected data to identify trends and anomalies.
  + **Process:** Analyze fuel consumption patterns, vehicle health metrics, and route data for actionable insights.

## Decision Making

* + **Objective:** Enable informed decisions for fuel optimization and vehicle maintenance.
  + **Process:** Generate alerts for low fuel levels, inefficient routes, or potential mechanical issues.

## Continuous Monitoring & Improvement

* + **Objective:** Maintain efficiency and adapt the system to evolving needs.
  + **Process:** Continuously monitor data trends, update algorithms, and integrate new sensors or features as needed.
  1. **Block Diagram**

Data

collection

Energy

saving

Data

cleaning

SMART

METER

Data

Visulaization

Data

analysis

Data

transform

# CHAPTER 3

**MODULES**

**1. Data Collection and Preprocessing Module**

Description: Cleans and prepares the dataset for analysis, including handling missing values, formatting timestamps, and detecting outliers.

Key Techniques:

pandas for data wrangling.

Detect and remove anomalies using the interquartile range (IQR) or z-scores.

**2. Statistical Analysis Module**

Description: Computes summary statistics (mean, median, standard deviation) to establish baseline energy usage patterns.

Key Techniques:

Aggregate data over timeframes (e.g., daily, weekly).

Analyze energy use by categories (appliance, time of day).

**3. Time Series Analysis Module**

Description: Identifies trends, seasonality, and deviations in energy usage.

Key Techniques:

Decompose time-series data (trend, seasonality, residual).

Forecast future energy use with ARIMA or Prophet models.

**4. Visualization Module**

Description: Creates visual representations of energy consumption patterns.

Key Techniques:

Line plots for trends, bar charts for categorical data, and heatmaps for correlations.

**5. Energy Efficiency Recommendations Module**

Description: Synthesizes findings into actionable recommendations for energy conservation.

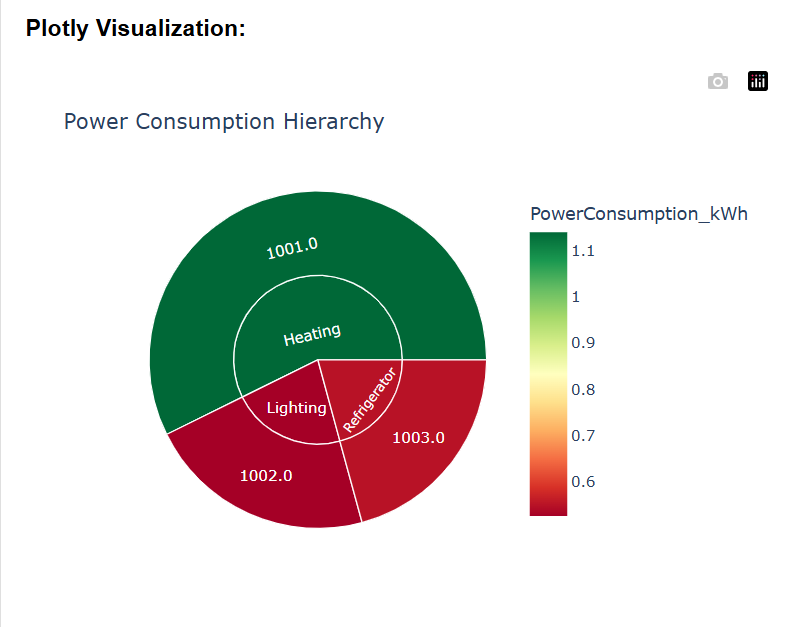
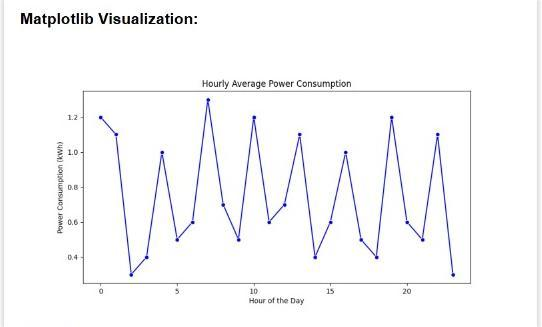
Key Techniques:

Generate personalized tips based on high-usage trends and anomalies.

# CHAPTER 4

# RESULTS AND DISCUSSION

**4.1 Results**

****

**4.2 Discussion**

The graph shows the **hourly average power consumption** throughout the day. The x-axis represents the **hour of the day**, while the y-axis represents **power consumption in kWh**.

1. **Fluctuating Pattern**: The power consumption varies significantly over the hours, showing a non-linear trend.
2. **Peak Hours**: The consumption reaches its highest levels at certain hours, such as around the 2nd, 10th, and 22nd hours.
3. **Lowest Usage**: The power consumption drops to its lowest points during specific hours, such as around the 4th and 15th hours.
4. **Insights**: These variations could be due to changes in human activity, such as peak working or resting times, or operational patterns in systems consuming power. The data suggests that power usage is dynamic and may benefit from scheduling energy-intensive tasks during off-peak times to optimize consumption.

# CHAPTER 5

# CONCLUSION

**5.1 Conclusion**

This project demonstrates the utility of Python's pandas library in analyzing smart meter data for energy efficiency. By leveraging robust preprocessing, statistical analysis, and visualization techniques, actionable insights were generated to empower users to reduce costs and environmental impact. The findings underscore the potential of data-driven approaches in promoting sustainable energy practices. Future work could incorporate machine learning for predictive modeling and real-time analytics.

**5.2 References**

* *Python for Data Analysis* by Wes McKinney – Comprehensive guide on using pandas for data manipulation.
* Energy.gov – Resources on energy-saving measures and smart meter technologies.
* Kaggle – Open datasets for energy consumption analysis.
* Matplotlib and Seaborn documentation – Guides for creating effective visualizations.

IEEE Xplore – Research papers on energy efficiency and smart meter analytics.

# APPENDIX

from flask import Flask, render\_template, request, redirect, url\_for

import pandas as pd

import os

import matplotlib.pyplot as plt

import seaborn as sns

import plotly.express as px

app = Flask(\_name\_)

UPLOAD\_FOLDER = 'uploads'

app.config['UPLOAD\_FOLDER'] = UPLOAD\_FOLDER

if not os.path.exists(UPLOAD\_FOLDER):

os.makedirs(UPLOAD\_FOLDER)

# Global variables to store uploaded dataset and file name

uploaded\_data = None

uploaded\_file\_name = None

@app.route('/', methods=['GET', 'POST'])

def index():

global uploaded\_data, uploaded\_file\_name

message = None # For status feedback

if request.method == 'POST':

# Check if the POST request has the file

if 'file' not in request.files:

message = "No file part in the request."

return render\_template('index.html', message=message)

file = request.files['file']

if file.filename == '':

message = "No file selected for uploading."

return render\_template('index.html', message=message)

# Save the file

file\_path = os.path.join(app.config['UPLOAD\_FOLDER'], file.filename)

file.save(file\_path)

# Load the file into a DataFrame

try:

if file.filename.endswith('.csv'):

uploaded\_data = pd.read\_csv(file\_path)

elif file.filename.endswith('.xlsx'):

uploaded\_data = pd.read\_excel(file\_path)

else:

message = "Unsupported file type. Please upload a CSV or Excel file."

return render\_template('index.html', message=message)

uploaded\_file\_name = file.filename

message = f"File '{uploaded\_file\_name}' uploaded successfully!"

except Exception as e:

message = f"Error processing the file: {e}"

return render\_template('index.html', message=message, uploaded\_file\_name=uploaded\_file\_name)

@app.route('/visualize', methods=['GET'])

def visualize():

global uploaded\_data

if uploaded\_data is None:

return "No data uploaded. Please upload a dataset first."

try:

# Convert Timestamp column to datetime

if 'Timestamp' in uploaded\_data.columns:

uploaded\_data['Timestamp'] = pd.to\_datetime(

uploaded\_data['Timestamp'], format='%d-%m-%Y %H.%M', errors='coerce'

)

# Add Hour column if Timestamp exists

if 'Timestamp' in uploaded\_data.columns:

uploaded\_data['Hour'] = uploaded\_data['Timestamp'].dt.hour

# Aggregation: Power Consumption by Hour

if 'Hour' in uploaded\_data.columns and 'PowerConsumption\_kWh' in uploaded\_data.columns:

hourly\_consumption = uploaded\_data.groupby('Hour')['PowerConsumption\_kWh'].mean().reset\_index()

# Create a Matplotlib plot

plt.figure(figsize=(10, 5))

sns.lineplot(x='Hour', y='PowerConsumption\_kWh', data=hourly\_consumption, marker='o', color='b')

plt.title("Hourly Average Power Consumption")

plt.xlabel("Hour of the Day")

plt.ylabel("Power Consumption (kWh)")

plot\_path = os.path.join("static", "hourly\_consumption.png")

plt.savefig(plot\_path)

plt.close()

else:

plot\_path = None

# Create an interactive Plotly visualization

if 'DeviceType' in uploaded\_data.columns and 'DeviceID' in uploaded\_data.columns:

fig = px.sunburst(

uploaded\_data, path=['DeviceType', 'DeviceID'], values='PowerConsumption\_kWh',

color='PowerConsumption\_kWh', color\_continuous\_scale='RdYlGn',

title="Power Consumption Hierarchy"

)

sunburst\_html = fig.to\_html(full\_html=False)

else:

sunburst\_html = "No DeviceType or DeviceID columns available for visualization."

except Exception as e:

return f"An error occurred during visualization: {e}"

return render\_template('index.html', plot\_path=plot\_path, sunburst\_chart=sunburst\_html)

if \_name\_ == '\_main\_':

app.run(debug=True)