COLLEGE CODE : 3105

COLLEGE NAME : DHANALAKSHMI SRINIVASAN COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT : B.E-COMPUTER SCIENCE AND ENGINEERING

STUDENT NM-ID : 0363df334a4681587d54907d02ad5697

ROLL NO : 310523104129

DATE : 07/05/2025

TECHNOLOGY-PROJECT NAME :

AI-ENERGY EFFICIENCY OPTIMIZATION

SUBMITTED BY,

SARAVANAN J

SARVEASWARAN S

SANDHIYA R

SHANIYA S F

SARAN R

Phase 5: Project Demonstration & Documentation

Title: AI-Energy Efficiency Optimization

Abstract

The integration of AI in energy efficiency optimization enhances resource management, reduces wastage, and ensures sustainable practices. AI-driven analytics, IoT-enabled monitoring, and automated control systems enable intelligent energy consumption while maintaining operational efficiency across various industries. This project leverages machine learning models to predict energy demand, optimize power distribution, and ensure secure data handling for maximum efficiency.

----------Index should be included with page number----------

1. Project Demonstration

Overview

The AI-driven energy optimization system will be demonstrated to highlight its ability to analyze energy consumption, predict patterns, and dynamically adjust operations for improved efficiency.

Demonstration Details

- **System Walkthrough:** Showcasing AI-based energy tracking, prediction models, and automated efficiency optimization.
- **AI Diagnosis Accuracy:** Demonstrating AI-driven energy demand forecasting and system adjustments to minimize wastage.
- **IoT Integration:** Live demonstration of real-time energy monitoring using IoT sensors and AI-powered decision-making.
- **Performance Metrics:** Highlighting system response time, energy savings, and cost efficiency achieved through AI algorithms.
- **Security & Privacy:** Ensuring data protection in energy management, emphasizing encryption protocols and secure transmission.

Outcome

Stakeholders will witness how AI-powered systems optimize energy consumption, reduce waste, and improve operational efficiency through real-time data analysis and predictive modeling.

2. Project Documentation

Overview

Comprehensive documentation outlining AI models, system architecture, operational workflows, and IoT integration strategies for optimized energy usage.

Documentation Sections

- **System Architecture:** Graphical representation of AI-driven energy optimization frameworks.
- **Code Documentation:** Explanation of AI algorithms used for demand forecasting and efficiency enhancement.
- **User Guide:** Instructions on utilizing AI-powered energy management tools for smart monitoring and automation.
- **Administrator Guide:** Guidelines for maintaining AI models, updating optimization parameters, and troubleshooting system performance.
- **Testing Reports:** Performance evaluations including energy consumption reductions, cost savings, and scalability assessments.

Outcome

The documentation will serve as a resource for developers, stakeholders, and system administrators to understand, refine, and expand AI-driven energy efficiency solutions.

3. Feedback and Final Adjustments

Overview

Insights from test users, energy analysts, and stakeholders will be gathered to refine AI algorithms for optimal energy efficiency performance.

Steps

- **Feedback Collection:** Gathering user observations on AI-driven energy predictions and optimization effectiveness.
- **Refinement:** Adjusting AI models based on feedback to improve accuracy, adaptability, and responsiveness.
- **Final Testing:** Running additional simulations and real-world tests to verify energy efficiency gains before final deployment.

Outcome

The refined AI system will ensure optimal energy distribution, reduced wastage, and improved performance across various environments.

4. Final Project Report Submission

Overview

A detailed report summarizing all phases of AI-powered energy optimization, including results, challenges, and future recommendations.

Report Sections

- **Executive Summary:** Overview of the AI-driven energy efficiency project and its impact.
- **Phase Breakdown:** Step-by-step analysis of AI model development, IoT integration, and security implementations.
- **Challenges & Solutions:** Discussion of obstacles in real-time predictions, scalability, and automation, along with strategies to overcome them.
- **Outcomes:** Summary of energy savings, efficiency improvements, and environmental benefits achieved through AI technologies.

Outcome

A comprehensive report will be submitted, showcasing the effectiveness of AI in optimizing energy consumption and sustainability practices.

5. Project Handover and Future Works

Overview

Final project delivery along with recommendations for scaling AI-driven energy efficiency systems for broader applications.

Handover Details

- **Next Steps:** Potential advancements in AI models, increased integration with smart grid technologies, and improved automation for long-term sustainability.

Outcome

The AI-driven energy optimization system will be handed over, with insights into future developments, enhancements, and global scalability.

Code and Progress

```
import random
import pandas as pd
from sklearn.ensemble import GradientBoostingRegressor
import joblib
from cryptography.fernet import Fernet
import time

class EnergyOptimizerAI:
    def __init__(self):
        self.model = GradientBoostingRegressor()

    def train_model(self, data):
        X = data[['temperature', 'occupancy', 'hour']]
        y = data['energy_usage']
        self.model.fit(X, y)
        joblib.dump(self.model, 'ai_energy_model.pkl')

    def predict_energy(self, input_features):
        model = joblib.load('ai_energy_model.pkl')
        return model.predict([input_features])[0]
```

```
def simulate_iot_sensor():
    return {
        'temperature': random.uniform(17, 34),
        'occupancy': random.randint(0, 20),
        'hour': random.randint(0, 23),
        'energy_usage': random.uniform(150, 2700)
    }

def create_dataset(size=250):
    return pd.DataFrame([simulate_iot_sensor() for _ in range(size)])

def encrypt_data(data, key):
    return Fernet(key).encrypt(data.encode())

def decrypt_data(encrypted, key):
    return Fernet(key).decrypt(encrypted).decode()
```

```
def chatbot(query):
    q = query.lower()
    if "optimize" in q:
        return "Reduce HVAC 2–4 PM, use motion lights."
    elif "usage" in q:
        return "Avg usage: 1125W. Try smart scheduling."
    elif "save" in q:
        return "Use efficient devices, run off-peak."
    else:
        return "Ask about energy saving or usage tips."

def test_performance(model, sample_input):
    start = time.time()
    for _ in range(100):
        model.predict_energy(sample_input)
    return round(time.time() - start, 3)

def main():
    print("AI-Energy Efficiency Optimization System Demonstration")
    print("------------------------------------------------------")
```

```
    data = create_dataset()
    ai_model = EnergyOptimizerAI()
    ai_model.train_model(data)
    print("Model trained with real-time IoT data.")

    test_input = [24.5, 6, 13]
    prediction = ai_model.predict_energy(test_input)
    print(f"Predicted Energy Usage: {prediction:.2f}W")

    response1 = chatbot("how do I optimize energy?")
    print(f"Chatbot: {response1}")

    response2 = chatbot("current energy usage")
    print(f"Chatbot: {response2}")

    key = Fernet.generate_key()
    encrypted = encrypt_data("Confidential Usage Data: 1180W", key)
    decrypted = decrypt_data(encrypted, key)

    print(f"Encrypted: {encrypted}")
    print(f"Decrypted: {decrypted}")
```

```
    latency = test_performance(ai_model, [21.0, 3, 18])
    print(f"System Response Time under Load: {latency} seconds")

main()
```

Output

```
AI-Energy Efficiency Optimization System Demonstration
------------------------------------------------------
Model trained with real-time IoT data.
Predicted Energy Usage: 1134.26W
Chatbot: Reduce HVAC 2–4 PM, use motion lights.
Chatbot: Avg usage: 1125W. Try smart scheduling.
Encrypted: b'gAAAAABlY...'
Decrypted: Confidential Usage Data: 1180W
System Response Time under Load: 0.162 seconds
```