COLLEGE CODE: 3105

COLLEGE NAME:DHANALAKSHMI SRINIVASAN COLLEGE OF TECHNOLOGY

DEPARTMENT:COMPUTER SCIENCE

STUDENT NM-ID:8B8E518B7321260A58E429D76520FB08

ROLL NO:310523104004

DATE:7/5/25

COMPLETED THE PROJECT NAMED AS
TECHNOLOGY-PROJECT NAME:ENERGY EFFICIENT OPTIMIZATION

SUBMITTED BY,
NAME :ADITYA AKSHAYA.M
MOBILE NO:8870137326

Energy Efficiency Optimization

Abstract:

The AI-Driven Energy Efficiency Optimization System aims to significantly reduce energy consumption in industrial and residential environments by utilizing artificial intelligence, IoT sensors, and smart automation. This final phase integrates predictive analytics, real-time energy monitoring, and secure data handling, ensuring scalable deployment and ERP integration. This document outlines the completed system demonstration, technical documentation, performance benchmarks, and testing results. The system is engineered to enhance sustainability by making data-driven decisions for optimizing energy usage. Screenshots, architectural diagrams, and code snippets are included for comprehensive understanding.

1. Project Demonstration

Overview:
The Energy Optimization System will be demonstrated to stakeholders, showing its AI-based decision-making, real-time sensor integration, and energy-saving automation.

Demonstration Details:

- System Walkthrough: Live interaction with the dashboard, showing real-time energy consumption and optimization suggestions.
- AI-Based Optimization: The AI model recommends appliance scheduling, lighting control, and HVAC adjustments for peak efficiency.
- IoT Integration: Data from smart meters, motion sensors, and environmental sensors is visualized and analyzed.
- Performance Metrics: Key metrics like energy saved (in kWh), system response time, and cost reductions will be demonstrated.
- Security Measures: Demonstration of encrypted data transmission and access control for energy data.

Outcome:
Stakeholders will see how the system reduces energy consumption while maintaining comfort and operational standards.

2. Project Documentation

Overview:
In-depth documentation covering system design, AI logic, sensor integration, and administrative tools.

Documentation Sections:

- System Architecture: Visual diagrams of AI pipelines, data flow from sensors to dashboards, and optimization loops.
- Code Documentation: Explanation of machine learning models, energy usage forecasting algorithms, and device APIs.
- User Guide: Steps for users to interpret energy data, receive suggestions, and apply recommendations.
- Administrator Guide: Tools for device calibration, system diagnostics, and energy profiling.
- Testing Reports: Load testing, accuracy evaluations, and simulations under different energy usage scenarios.

Outcome:
The documentation supports future scaling, user onboarding, and system troubleshooting.

3. Feedback and Final Adjustments

Overview:
Feedback gathered during live demo and trials informs the final refinements.

Steps:

- **Feedback Collection:** User feedback from demo sessions, test environments, and stakeholder surveys.
- **Refinement:** Fixes to sensor calibration issues, AI tuning, and user interface improvements.
- **Final Testing:** Performance revalidation to ensure stability and energy optimization accuracy.

Outcome:
System fully optimized for deployment across varied energy consumption environments.

4. Final Project Report Submission

Overview:
A full report summarizing all project phases, technical milestones, and performance outcomes.

Report Sections:

Executive Summary: Project goals and quantifiable achievements (e.g., 20–30% energy savings).

Phase Breakdown: Description of AI training, sensor network development, and system integrations.

Challenges & Solutions: Key issues like inconsistent sensor data or optimization conflicts and how they were resolved.

Outcomes: Current system capabilities and successful implementation results.
Clear documentation of project success and its potential for real-world impact.

5. Project Handover and Future Works

Overview:
Transfer of deliverables and roadmap for future enhancement.

Handover Details:

Next Steps: Suggestions include deeper AI personalization, support for renewable sources, and integration with utility APIs.

Outcome:
Project transferred with guidelines for long-term support and upgrade paths.

PROGRAM CODE:

```
import random
import numpy as np
def energy_model(params):
    x, y = params
energy_use = (x - 3)**2 + (y - 2)**2 + np.sin(3 * x) + np.cos(2 * y)
    return energy_use
POP_SIZE = 50
NUM_GENERATIONS = 100
MUTATION_RATE = 0.1
NUM_PARAMS = 2
BOUNDS = [(-10, 10), (-10, 10)]  # parameter bounds
def random_individual():
    return [random.uniform(*BOUNDS[i]) for i in range(NUM_PARAMS)]
def mutate(ind):
    i = random.randint(0, NUM_PARAMS - 1)
    ind[i] += random.uniform(-1, 1)
    ind[i] = max(min(ind[i], BOUNDS[i][1]), BOUNDS[i][0])
    return ind
def crossover(parent1, parent2):
    return [(p1 + p2) / 2 for p1, p2 in zip(parent1, parent2)]
population = [random_individual() for _ in range(POP_SIZE)]
for generation in range(NUM_GENERATIONS):
population.sort(key=energy_model)
    best = population[0]
    print(f"Gen {generation}: Best Params = {best}, Energy = {energy_model(best):.4f}")
parents = population[:POP_SIZE // 2]
new_population = parents[:]
```

```
 while len(new_population) < POP_SIZE:
        p1, p2 = random.sample(parents, 2)
        child = crossover(p1, p2)
        if random.random() < MUTATION_RATE:
            child = mutate(child)
        new_population.append(child)
    population = new_population
best_solution = population[0]
print(f"\n🔍 Optimal Parameters Found: {best_solution}")
print(f"⚡ Minimum Energy Use: {energy_model(best_solution):.4f}")
```