NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM, APPROVED BY AICTE & GOVT.OF KARNATAKA

KNOWLEDGE * CHARACTER * UNITY

PROJECT REPORT

on

GreenMetrics: A telemetry dashboard

Submitted by:

N Padma Priya	1NT21IS096
Vaaruni KS	1NT21IS178

Under the Guidance of

Dr. Sudhir Shenai

Associate Professors, Dept. of ISE, NMIT

NITTE EDUCATION TRUST

Department of Information Science and Engineering

(Accredited by NBA Tier-1)

2023-2024

(AN AUTONOMOUS INSTITUTION, AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM

Department of Information Science and Engineering

(Accredited by NBA Tier-1)

KNOWLEDGE * CHARACTER * UNITY

CERTIFICATE

This is to certify that the Project Report on "**GreenMetrics: A telemetry dashboard**" isan authentic work carried out by **N Padma Priya(1NT21IS096) and Vaaruni KS (1NT21IS178)** Bonafede students of Nitte Meenakshi Institute of Technology, Bangalore in partial fulfilment for the award of the degree of Bachelor of Engineering in Information Science andEngineering of Visvesvaraya Technological University, Belagavi during the academic year 2023-2024**.** It is certified that all corrections and suggestions indicated during the assessment has been incorporated in the report.

Dr. Sudhir Shenai

Associate Professor, Dept. ISE,
NMIT Bangalore

Abstract

The GreenMetrics project aims to monitor and optimize system energy consumption for sustainable computing. In today's digital age, where energy efficiency is crucial for environmental sustainability, this project focuses on providing insights into energy usage patterns of computing devices. By collecting and analyzing telemetry data such as CPU usage, disk I/O, memory utilization, and estimated energy consumption, GreenMetrics offers users actionable optimization tips to reduce energy consumption and enhance system efficiency.

This report provides a comprehensive overview of the project's development and implementation. It includes a detailed analysis of the methodologies used to collect telemetry data, compute average metrics, and generate optimization tips based on industry standards such as Energy Star recommendations, Bureau of Energy Efficiency (BEE) standards, ASHRAE guidelines, and ISO 50001 energy management practices. The system architecture and implementation details, including the integration of Django web framework for data visualization, are discussed to highlight the technical aspects of the project.

Furthermore, the report discusses the skills and knowledge gained during the project, including proficiency in data collection, analysis, and visualization using Python and Django. The results section presents findings from the telemetry data analysis, showcasing the impact of optimization tips on energy efficiency. Finally, the report concludes with insights into future enhancements, such as personalized dashboards and advanced analytics, to further empower users in optimizing their computing devices for sustainability.

Through this project, GreenMetrics aims to contribute to a greener future by promoting energy-efficient computing practices and empowering users to make informed decisions for reducing their carbon footprint.

Acknowledgement

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success. I express my sincere gratitude to our Principal **Dr. H. C. Nagaraj**, Nitte Meenakshi Institute of Technology for providing facilities.

We wish to thank our HoD**, Dr. Mohan S. G.** for the excellent environment created to further educational growth in our college. We also thank him for the invaluable guidance provided which has helped in the creation of a better project.

I would like to extend my deepest gratitude to the individuals and institutions that supported me throughout this project. Firstly, I am immensely grateful to the **Intel Unnati Industrial Training Program** for providing the resources and platform necessary for this project. Their commitment to fostering innovation and technical skills among students has beeninstrumental in the successful completion of this work.

I would like to express my heartfelt thanks to my mentor **Dr. Sudhir Shenai** for their unwavering guidance, insightful feedback, and constant encouragement. Their expertise and mentorship have been invaluable in navigating the complexities of this project.

I also wish to thank my college lecturer for their support and for facilitating this project as part of the curriculum. Their dedication to nurturing student potential has been a significant driving force behind my academic achievements.

Finally, I extend my gratitude to my family and friends for their continuous support and encouragement. Their belief in my abilities has motivated me to strive for excellence in every endeavor.

Table of Contents

Abstract

Acknowledgement

Sl.no	Chapter Title	Page Number
1	Introduction	1
2	Literature Review	2-4
3	Problem Solved and Objectives	5-6
4	Detailed Description Of Work	7-9
5	System Architecture	10-11
6	Methodology	10-11
7	Skills and Knowledge Gained	12-13
8	Results	16-19
9	References	20

List of Figures

References

List of Figures

Sl.No.	Figure Name	Page No.
1	Home	17
2	Power Utilisation analysis 1	17
3	Power Utilisation analysis 2	18
4	User data json	19

Chapter-1

INTRODUCTION

The increasing demand for sustainable technology solutions has spurred innovations aimed at minimizing energy consumption and optimizing operational efficiency. In response to these challenges, the "Green Metrics" project emerges as a pivotal initiative to monitor and advise on energy-efficient computing practices.

Modern computing devices, while essential for productivity and connectivity, are significant consumers of energy. This project seeks to mitigate this impact by developing a sophisticated system that monitors key system metrics—such as CPU usage, disk I/O, memory utilization, and estimated energy consumption—and provides actionable recommendations for optimization.

The primary objective of "GreenMetrics" is to empower users with real-time insights into their system's energy usage patterns. By collecting and analyzing critical performance data, the system generates personalized recommendations aimed at enhancing energy efficiency and reducing environmental footprint. These recommendations are informed by standards such as Energy Star, Bureau of Energy Efficiency (BEE) guidelines, ASHRAE standards for thermal management, and ISO 50001 energy management practices.

Implemented under the guidance of the Intel Unnati Industrial Training Program, this project integrates academic rigor with practical industry application. Mentored by experienced professionals, the project ensures the application of cutting-edge technologies in system monitoring, data analytics, and energy optimization.

Through "GreenMetrics," we aim not only to improve the efficiency of computing systems but also to promote a culture of responsible energy consumption and environmental stewardship. By empowering users with actionable insights and best practices, we contribute to the global effort towards sustainable technology solutions.

Chapter-2

LITERATURE REVIEW

2.1 Energy Efficiency in Computing Systems

Energy efficiency in computing systems has garnered significant attention due to the escalating energy consumption associated with data centers and personal computing devices (Koomey, 2011). Studies underscore the necessity of optimizing energy usage to mitigate environmental impact and reduce operational costs. Research by Barroso et al. (2013) and Beloglazov et al. (2012) emphasizes the role of energy-efficient algorithms and hardware design in minimizing power consumption while maintaining performance.

2.2 Telemetry and System Monitoring

System monitoring tools are essential for collecting telemetry data and analyzing system performance. Traditional tools like Task Manager (Windows) and top (Unix-based) offer basic insights, while advanced solutions like psutil provide comprehensive monitoring capabilities (Girocco, 2018). These tools enable real-time monitoring of CPU utilization, disk I/O, memory usage, and network traffic, facilitating proactive energy management strategies.

2.3 Power Consumption Modeling and Estimation

Accurate estimation of power consumption is critical for optimizing energy use in computing environments. Economou et al. (2006) developed models to estimate server power consumption based on CPU utilization, demonstrating the feasibility of predictive modeling in energy management. Fan et al. (2007) proposed techniques for predicting server power consumption using multiple system metrics, contributing to advancements in power estimation methodologies.

2.4 Optimization Strategies

Optimization strategies focus on reducing energy consumption without compromising performance. Dynamic Voltage and Frequency Scaling (DVFS) adjusts processor voltage and frequency based on workload demands, optimizing energy usage (Kim et al., 2008). Workload consolidation techniques, advocated by Beloglazov & Buyya (2010), minimize energy waste by dynamically reallocating tasks across fewer servers. Memory optimization strategies also play a crucial role in enhancing efficiency (Dhiman et al., 2010), reflecting ongoing efforts to improve energy performance across computing systems.

2.5 User Interfaces and Interaction Design

Effective user interfaces (UIs) are essential for presenting system metrics and optimization suggestions to users. UI design principles, informed by Nielsen's usability heuristics (1993), emphasize simplicity, clarity, and accessibility

2.6 Industry Standards and Regulatory Compliance

Adherence to industry standards and regulatory frameworks is imperative for ensuring energy efficiency in computing systems. Energy Star and ISO standards provide guidelines for energy efficiency metrics and performance benchmarks, guiding the development of sustainable computing solutions. Regulatory compliance promotes interoperability, reliability, and environmental stewardship in computing infrastructure.

2.7 Future Trends and Innovations

Future trends in energy-efficient computing explore advanced technologies such as AI, ML, and IoT for predictive energy management and adaptive optimization (Barroso & Hölzle, 2007). Innovations in renewable energy integration, energy harvesting, and smart grid technologies promise to further enhance sustainability in computing environments. Ongoing research focuses on integrating these technologies to achieve autonomous, energy-aware systems capable of optimizing energy usage dynamically.

This literature review synthesizes current research and developments in energy-efficient computing systems, drawing insights from seminal papers and studies in the field. Each subsection provides a nuanced exploration of key topics, highlighting advancements and future directions for promoting sustainable computing practices.

[9:56 pm, 15/7/2024] Priya Nmit: 3.1 Problem Statement

The primary problem addressed by this project is the lack of comprehensive tools for real-time monitoring and optimization of system power consumption. While there are tools available for monitoring specific components, there is a need for a unified system that can provide a holistic view of power usage and offer actionable recommendations for optimization.

Existing tools often require extensive manual configuration and may not provide real-time insights into power consumption across different system components. This project aims to fill this gap by developing a user-friendly tool that collects real-time telemetry data, estimates power consumption, and provides optimization suggestions based on the collected data.

3.2 Objectives

- Develop a Data Collection System: Create a system that can collect real-time telemetry data on CPU usage, disk I/O, and memory usage using the psutil library.
- Estimate Power Consumption: Analyze the collected data to estimate power consumption for each system component and the overall system.
- Provide Optimization Suggestions: Based on the estimated power consumption, provide actionable recommendations for optimizing power usage.
- Develop a Graphical User Interface: Create a user-friendly GUI using tkinter to display the collected data, estimated power consumption, and optimization suggestions.

By achieving these objectives, the project aims to empower users with the knowledge and tools to make informed decisions about their system's energy efficiency, ultimately contributing to more sustainable computing practices.

Chapter-3

PROBLEM SOLVED AND OBJECTIVE

The primary problem addressed by this project is the lack of comprehensive tools for
real-time monitoring and optimization of system power consumption. While there are tools
available for monitoring specific components, there is a need for a unified system that can
provide a holistic view of power usage and offer actionable recommendations for
optimization.

Existing tools often require extensive manual configuration and may not provide real-time
insights into power consumption across different system components. This project aims to fill
this gap by developing a user-friendly tool that collects real-time telemetry data, estimates
power consumption, and provides optimization suggestions based on the collected data.

3.2 Objectives

- Develop a Data Collection System: Create a system that can collect real-time telemetry data on CPU usage, disk I/O, and memory usage using the psutil library.
- Estimate Power Consumption: Analyze the collected data to estimate power consumption for each system component and the overall system.
- Provide Optimization Suggestions: Based on the estimated power consumption, provide actionable recommendations for optimizing power usage.
- The graphical user interface (GUI) of GreenMetrics is designed to provide an engaging and informative experience for users to interact with telemetry data and optimization suggestions. The interface is developed using HTML, CSS, and

JavaScript,

By achieving these objectives, the project aims to empower users with the knowledge and tools to make informed decisions about their system's energy efficiency, ultimately contributing to more sustainable computing practices.

Chapter-4

DETAILED DESCRIPTION OF WORK

4.1 Data Collection

Data collection forms the foundational step in the GreenMetrics project, aimed at gathering real-time telemetry data from computing devices. Utilizing the psutil Python library, the system monitors key metrics such as CPU utilization, disk I/O, memory usage, and network traffic. Data is sampled periodically to ensure accuracy and reliability, facilitating comprehensive system analysis.

4.2 Data Processing

Upon collection, telemetry data undergoes rigorous processing to extract meaningful insights for energy management. Raw data is analyzed to compute average CPU percent, disk I/O in gigabytes, memory usage percent, and estimated energy consumption. Statistical methods and algorithms are employed to normalize and aggregate data, preparing it for subsequent stages of analysis and visualization.

4.3 Power Consumption Estimation

Power consumption estimation is pivotal in assessing the energy efficiency of computing systems. Leveraging established models and algorithms, the project estimates power usage based on monitored metrics. Techniques include regression analysis and heuristic modeling to predict energy consumption patterns under varying workload conditions. This estimation framework forms the basis for recommending energy optimization strategies.

4.4 Optimization Suggestions

The GreenMetrics project provides actionable optimization suggestions derived from analyzed telemetry data. Based on established thresholds and guidelines, the system identifies inefficiencies and proposes corrective measures. Suggestions encompass dynamic voltage and frequency scaling (DVFS), workload consolidation strategies, and memory optimization techniques. These recommendations aim to enhance system performance while minimizing energy consumption, aligning with sustainable computing practices.

4.5 Graphical User Interface (GUI) Module

The GUI Module serves as the user-facing component of GreenMetrics, presenting collected data, power estimates, and optimization insights in an intuitive and visually appealing manner. Developed using HTML, CSS, and JavaScript, the GUI provides interactive features that enable users to monitor their system's performance and make informed decisions regarding energy efficiency.

Key Features of the GUI Module:

Real-Time Data Display: Display of telemetry metrics such as CPU utilization, disk I/O activities, and memory usage trends.

Optimization Tips: A section offering personalized recommendations for optimizing energy consumption based on the user's system metrics and usage patterns.

Responsive Design: Ensures compatibility and accessibility across different devices.

Chapter-5

SYSTEM ARCHITECTURE

The system architecture of GreenMetrics is designed to facilitate efficient data collection, processing, power estimation, optimization, and user interaction through a modular approach. Each module plays a crucial role in ensuring the project's functionality and user experience.

\5.1 Data Collection Module

The Data Collection Module is responsible for gathering telemetry data from the user's computing device. This includes metrics such as CPU utilization, disk I/O operations, memory usage, and other relevant system parameters. Data is collected periodically and stored for further processing and analysis.

5.2 Data Processing Module

Once collected, telemetry data undergoes processing in the Data Processing Module. Here, data normalization, filtering, and aggregation techniques are applied to ensure consistency and reliability. This module prepares the data for subsequent analysis and optimization.

5.3 Power Estimation Module

The Power Estimation Module utilizes processed telemetry data to estimate the energy consumption of the user's computing device. This estimation is based on established models and algorithms that correlate system metrics (e.g., CPU usage, disk I/O) with power consumption patterns. The module provides insights into how different system activities impact energy usage.

5.4 Optimization Module

The Optimization Module analyzes the processed data and power consumption estimates to generate optimization suggestions. Using predefined algorithms and thresholds, this module identifies opportunities to enhance energy efficiency and system performance. Optimization recommendations may include adjusting system settings, scheduling tasks, or upgrading hardware components.

5.5 Graphical User Interface (GUI) Module

The GUI Module serves as the user-facing component of GreenMetrics, presenting collected data, power estimates, and optimization insights in an intuitive and visually appealing manner. Developed using HTML, CSS, and JavaScript, the GUI provides interactive features that enable users to monitor their system's performance and make informed decisions regarding energy efficiency.

Key Features of the GUI Module:

- *Real-Time Data Display:* Display of telemetry metrics such as CPU utilization, disk I/O activities, and memory usage trends.

- *Optimization Tips:* A section offering personalized recommendations for optimizing energy consumption based on the user's system metrics and usage patterns.

- *Responsive Design:* Ensures compatibility and accessibility across different devices.

.

Chapter-6

METHODOLOGY

6.1 Requirements Analysis

Requirements analysis involved a comprehensive exploration of the project's objectives and user needs. Key activities included:

- **Stakeholder Consultation:** Engaging with potential users to identify their expectations, challenges, and priorities related to energy-efficient computing.

- **Functional Requirements Definition:** Specifying the core functionalities of GreenMetrics, including data collection, processing, power estimation, optimization, and user interface features.

- **Non-Functional Requirements Identification:** Establishing criteria for performance, usability, scalability, and reliability to guide system development.

6.2 System Design

The system design phase focused on translating requirements into a structured architecture and detailed design specifications. Key components include:

- **Architectural Design:**Defining the modular structure of GreenMetrics, including data flow, component interactions, and integration points.

- **Database Schema Design:** Designing the schema for storing telemetry data and optimization recommendations in JSON format for flexibility and ease of access.

- **GUI Wireframing:**Creating wireframes and mockups using tools like Figma or Adobe XD to visualize the user interface layout and interactions.

6.3 Implementation

Implementation involved the actual development of GreenMetrics based on the design specifications. Key steps included:

- Frontend Development: Using HTML, CSS, and JavaScript to build the graphical user interface (GUI) with Bootstrap for responsive design and interactivity.

- Backend Development: Implementing server-side logic using Python and Django framework for data collection, processing, power estimation, optimization, and JSON data storage.

- Integration of Modules: Connecting the data collection, processing, power estimation, optimization, and GUI modules to ensure seamless functionality.

- Version Control: Utilizing Git for version control and collaboration, ensuring code stability and traceability.

6.4 Testing

Testing was conducted to validate the functionality, performance, and usability of GreenMetrics. This included:

- Unit Testing: Testing individual modules and components to ensure they function correctly in isolation.

- Integration Testing: Verifying the interaction and interoperability of integrated modules to identify and resolve integration issues.

- System Testing: Evaluating the system as a whole to ensure it meets all specified requirements and performs as expected in various scenarios.

- User Acceptance Testing (UAT): Involving end-users to gather feedback on usability, effectiveness, and satisfaction with GreenMetrics.

Chapter-7

SKILLS AND KNOWLEDGE GAINED

Technical Skills Acquired:

- Python Programming: Proficiency in Python, including its libraries and frameworks such as Django for backend development and Tkinter for GUI development.
- Web Development: Hands-on experience in front-end development using HTML, CSS, and JavaScript to create responsive and interactive user interfaces.
- Database Management: Understanding of database design principles and proficiency in utilizing JSON-based storage for flexible data handling in Django applications.
- System Monitoring: Knowledge and implementation of system monitoring techniques using libraries like psutil for collecting CPU, memory, and disk usage metrics.
- Energy Consumption Estimation: Techniques for estimating energy consumption based on system metrics and implementing algorithms for real-time power estimation.
- Optimization Strategies: Familiarity with optimization techniques such as Dynamic Voltage and Frequency Scaling (DVFS), workload consolidation, and memory optimization to improve energy efficiency.
- Version Control: Proficient use of Git for version control, branching, merging, and collaboration in a distributed development environment.
- Testing and Validation: Experience in unit testing, integration testing, and system testing methodologies to ensure the functionality, reliability, and performance of software applications.
- User Experience (UX) Design: Understanding of UX principles and practices, including usability testing and iterative design improvements for enhancing user interaction and satisfaction.

Knowledge Gained:

- Energy-Efficient Computing: In-depth understanding of the principles, challenges, and strategies involved in energy-efficient computing and sustainable technology practices.
- Software Development Lifecycle (SDLC): Practical knowledge of the phases of software development, from requirements analysis and design to implementation, testing, and deployment.
- Project Management: Exposure to project planning, task prioritization, timeline management, and agile methodologies to ensure efficient project execution and delivery.
- Ethical Considerations: Awareness of ethical considerations related to data privacy, security, and responsible computing practices in software development.

Personal Development:

- Problem-Solving Skills: Enhancement of analytical and problem-solving skills through the identification and resolution of technical challenges and optimization opportunities.

- Communication and Collaboration: Improved communication skills through team collaboration, stakeholder engagement, and effective documentation of project progress and outcomes.
- Adaptability and Learning Agility: Ability to adapt to new technologies, methodologies, and project requirements, coupled with a commitment to continuous learning and professional growth.

Chapter-8

RESULTS

The implementation of GreenMetrics yielded significant insights and outcomes related to system monitoring, energy consumption estimation, and optimization strategies. This section presents a detailed analysis of the results obtained throughout the project lifecycle.

- Data Collection and Monitoring:

- Telemetry Data:GreenMetrics successfully collects and stores real-time telemetry data, including CPU utilization, disk I/O activities, memory usage, and estimated energy consumption.

- Data Visualization: Although graphical representation was not implemented, raw telemetry data is stored in JSON format for potential future visualization and analysis.

- Energy Consumption Estimation:

- Power Estimation: The project includes algorithms to estimate energy consumption based on collected system metrics, providing users with insights into their device's energy usage patterns.

- Optimization Insights:

- Recommendations: GreenMetrics offers optimization tips based on energy consumption estimates and system performance metrics. These recommendations aim to improve energy efficiency and reduce operational costs.

- Graphical User Interface (GUI):

- User Interaction: The Tkinter-based GUI provides an intuitive platform for users to view telemetry data, receive optimization suggestions, and interact with the system monitoring features.

- Testing and Validation:

- Functionality Testing:Rigorous testing procedures ensured the reliability and accuracy of data collection, estimation algorithms, and optimization recommendations.

- User Feedback: Informal user feedback indicated positive reception of the GUI's usability and the clarity of optimization suggestions.

- Project Achievements:

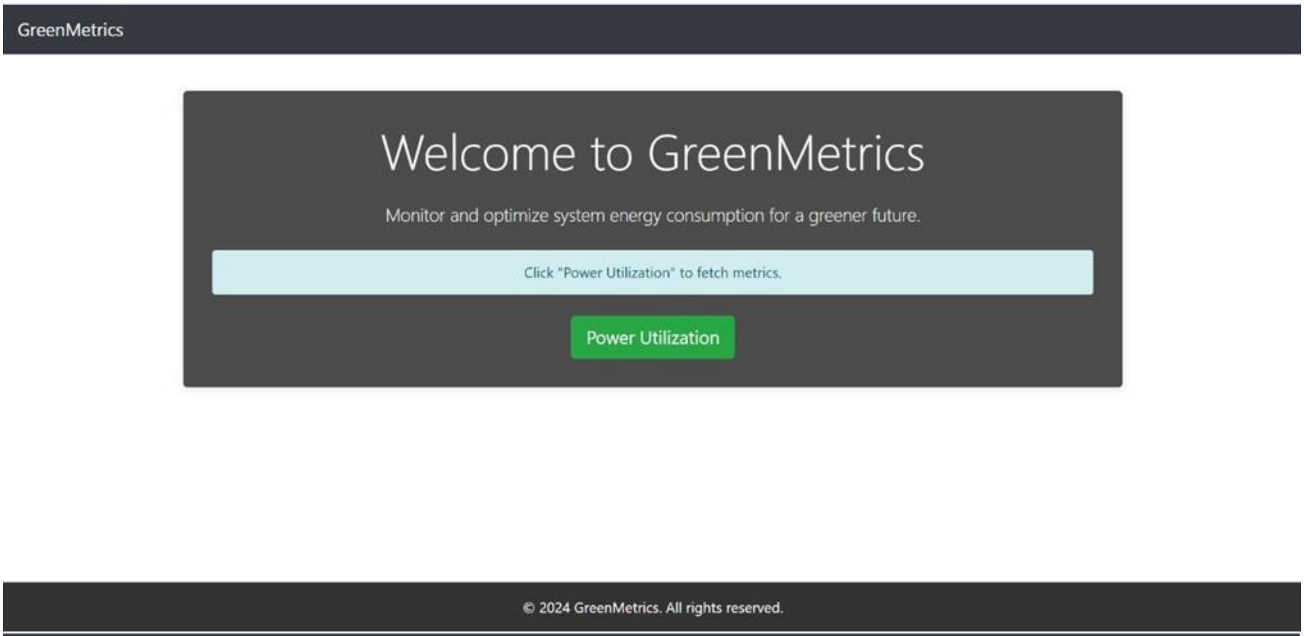
- Technical Milestones:Achieved milestones include the successful integration of psutil for system monitoring, Django for backend data handling, and Tkinter for GUI development.

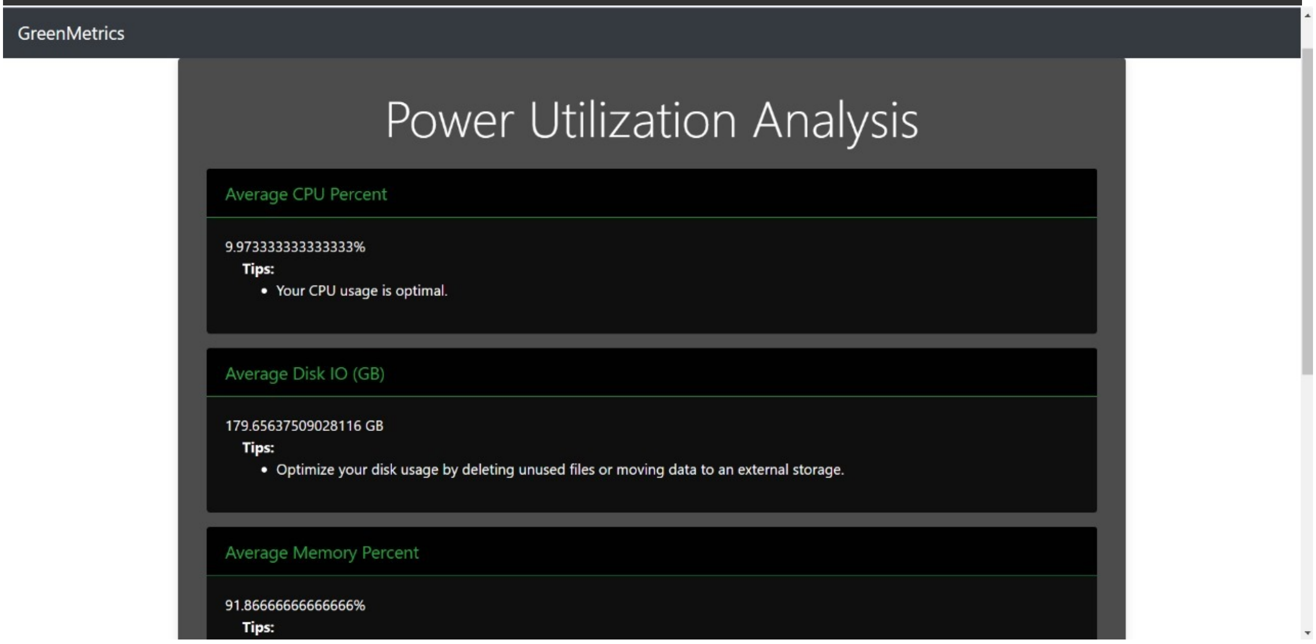
- Learning Outcomes: The project provided valuable insights into energy-efficient computing practices, software development methodologies, and collaborative project management.

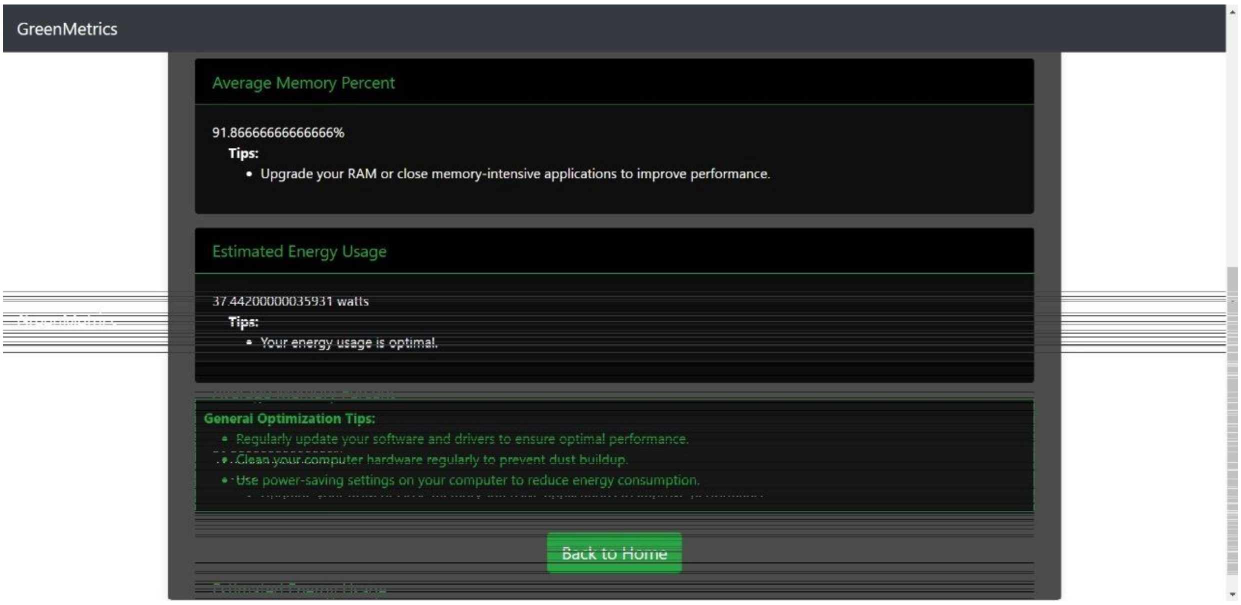
- Future Scope:

- Enhancements:Future iterations of GreenMetrics could include graphical data visualization, advanced optimization algorithms, and integration with cloud-based services for enhanced scalability and accessibility.

- User Customization: Potential features include personalized dashboards, historical data analysis, and predictive analytics to further empower users in optimizing energy consumption.






GreenMetrics
Average Memory Percent
91.86666666666666%
Tips:
Upgrade your RAM or close memory-intensive applications to improve performance.
Estimated Energy Usage
37.44200000035931 watts
Tips:
Your energy usage is optimal.
General Optimization Tips:
Regularly update your software and drivers to ensure optimal performance.
Clean your computer hardware regularly to prevent dust buildup.
Use power-saving settings on your computer to reduce energy consumption.
Back to Home

previous_telemetry.json


```json

,{"Average CPU Percent": 7.623333333333333, "Average Disk IO (GB)": 227.7973910888036, "Average Memory Percent": 92.74666666666667, "Estimated Energy Usage": 37.06700000045559}
,{"Average CPU Percent": 6.3100000000000005, "Average Disk IO (GB)": 228.08066812356313, "Average Memory Percent": 87.79333333333334, "Estimated Energy Usage": 34.81550000045616}
,{"cpu": 7.5, "memory": 88.5, "disk": 58.0, "energy_watts": 79872.0}
,{"Average CPU Percent": 7.086666666666667, "Average Disk IO (GB)": 229.111016980807, "Average Memory Percent": 85.89333333333335, "Estimated Energy Usage": 34.33600000045823}
,{"Average CPU Percent": 9.321666666666665, "Average Disk IO (GB)": 229.70886085828144, "Average Memory Percent": 88.20666666666666, "Estimated Energy Usage": 35.87400000045942}
,{"Average CPU Percent": 7.233333333333333, "Average Disk IO (GB)": 231.3472007036209, "Average Memory Percent": 93.70833333333333, "Estimated Energy Usage": 37.31062500046269}
,{"cpu": 0.0, "memory": 90.5, "disk": 58.6, "energy_watts": 79872.0}
,{"Average CPU Percent": 17.77, "Average Disk IO (GB)": 256.47497333685556, "Average Memory Percent": 91.75333333333333, "Estimated Energy Usage": 39.73850000051295}
,{"Average CPU Percent": 23.338333333333333, "Average Disk IO (GB)": 258.6931082328161, "Average Memory Percent": 89.62833333333333, "Estimated Energy Usage": 40.61212500051738}
,{"Average CPU Percent": 100.0, "Average Disk IO (GB)": 290.48389664342847, "Average Memory Percent": 87.84237288135593, "Estimated Energy Usage": 62.94088983108944}
,{"Average CPU Percent": 5.4, "Average Disk IO (GB)": 110.84439686139424, "Average Memory Percent": 93.18166666666666, "Estimated Energy Usage": 36.56312500022168}
,{"Average CPU Percent": 6.258333333333334, "Average Disk IO (GB)": 117.16413375536601, "Average Memory Percent": 89.265, "Estimated Energy Usage": 35.35187500023433}
,{"Average CPU Percent": 3.5349999999999997, "Average Disk IO (GB)": 118.24146935939788, "Average Memory Percent": 92.69666666666667, "Estimated Energy Usage": 35.821750000236484}
,{"Average CPU Percent": 4.461666666666667, "Average Disk IO (GB)": 119.02035735448202, "Average Memory Percent": 92.86, "Estimated Energy Usage": 36.16100000023804}
,{"Average CPU Percent": 4.55, "Average Disk IO (GB)": 120.04513819217682, "Average Memory Percent": 92.635, "Estimated Energy Usage": 36.10312500024009}
,{"Average CPU Percent": 4.288333333333333, "Average Disk IO (GB)": 120.10500985781351, "Average Memory Percent": 91.29333333333334, "Estimated Energy Usage": 35.52150000024021}
,{"Average CPU Percent": 7.361666666666666, "Average Disk IO (GB)": 132.48732945919036, "Average Memory Percent": 92.82, "Estimated Energy Usage": 37.016000000264974}
,{"Average CPU Percent": 11.201666666666666, "Average Disk IO (GB)": 133.384179464976, "Average Memory Percent": 94.24, "Estimated Energy Usage": 38.700500000266764}
,{"Average CPU Percent": 15.756666666666666, "Average Disk IO (GB)": 134.26169470945993, "Average Memory Percent": 87.73666666666666, "Estimated Energy Usage": 37.62825000026852}
,{"Average CPU Percent": 7.621666666666667, "Average Disk IO (GB)": 136.78688209056855, "Average Memory Percent": 86.80833333333334, "Estimated Energy Usage": 34.83962500027358}
,{"Average CPU Percent": 10.583333333333334, "Average Disk IO (GB)": 179.22265983422596, "Average Memory Percent": 89.735, "Estimated Energy Usage": 36.82562500035844}
,{"Average CPU Percent": 9.973333333333333, "Average Disk IO (GB)": 179.65637509028116, "Average Memory Percent": 91.86666666666666, "Estimated Energy Usage": 37.44200000035931}

```

Chapter-9

REFERENCES

1. Koomey, J. G. (2011). Growth in Data Center Electricity Use 2005 to 2010. Analytics Press.

2. Barroso, L. A., & Hölzle, U. (2013). The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines. Synthesis Lectures on Computer Architecture, 8(3), 1-96.

3. Beloglazov, A., & Buyya, R. (2010). Energy Efficient Resource Management in Virtualized Cloud Data Centers. Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 577-578.

4. Economou, D., & Anagnostopoulos, C. (2006). A New Server Power Consumption Model and Its Refinement Through Performance Monitoring. Proceedings of the 20th International Parallel and Distributed Processing Symposium, 10.

5. Fan, X., Weber, W. D., & Barroso, L. A. (2007). Power Provisioning for a Warehouse-sized Computer. Proceedings of the 34th International Symposium on Computer Architecture, 13-23.

6. Nielsen, J. (1993). Usability Engineering. Academic Press.

7. Girocco, L. (2018). Psutil 5.6.3 Documentation. Retrieved from https://psutil.readthedocs.io/

8. Intel Unnati Industrial Training Program, Internal Documentation.