IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

Green Software Tracker for Web Applications Using AI

Shivani Mandlik[1], Korade Prajkta[2], Dhamale Sneha[3], Prof Jayshree Bhagat[4]
[1] Final Year B.E. Computer Engineering Student
Department of Computer Engineering,
Jaihind College of Engineering, Kuran, Maharashtra, India.

Abstract : *Web applications consume computing resources such as CPU, memory, and network bandwidth during request processing. High resource consumption can increase energy usage and contribute to carbon emissions. This research proposes a Green Software Tracker for Web Applications Using AI to monitor resource utilization and estimate the environmental impact of web applications. The proposed system uses Python and the psutil library to collect process-level CPU and memory data at defined intervals. It aims to estimate energy consumption using a documented estimation method and calculate associated carbon emissions using an appropriate electricity emission factor. A dashboard is planned to display resource usage, estimated emissions, a Green Score, and recommendations for improving software efficiency. The current recommendation approach is rule-based. Further implementation and experimental evaluation are required to validate the estimation method and assess the system's performance.*

Key Words: Green Software, Web Applications, Energy Consumption, Carbon Emissions, Resource Monitoring, Sustainable Computing.

1. INTRODUCTION

Web applications have become an important part of everyday activities. People use them for online banking, education, shopping, communication, business operations, and many other services. These applications process user requests through several computing operations, including data processing, memory allocation, database communication, and network transfer. Each operation uses computing resources. The amount of resource usage can vary according to the application's design, workload, and execution environment.

Energy consumption has become an important consideration in software development. When an application uses processor time, memory, and network resources, the computer or server running that application requires electrical power. Inefficient operations, repeated requests, and unnecessary processing may increase resource usage. Therefore, developers need ways to observe application behaviour and understand its possible energy impact. Research on software energy measurement also identifies challenges in estimating energy consumption accurately and consistently [1], [2].

Existing approaches to software energy assessment use different measurement methods and tools. The Green Software Measurement Model presents a reference framework for evaluating software resource use and energy efficiency [1]. Research on website energy estimation also discusses the difficulty of obtaining reliable estimates and the importance of considering processor activity when assessing energy use [2]. These studies indicate that software resource monitoring can support energy assessment, but the estimation method and its limitations must be clearly defined.

The proposed Green Software Tracker for Web Applications Using AI is a software-based system designed to monitor selected resource indicators while a web application is running. It collects CPU utilization, memory consumption, and network activity using Python and the psutil library. The library provides system and process information that can be accessed through Python programs [3]. The collected measurements will be used by the proposed energy estimation method. The system will then estimate the related carbon dioxide (CO_2) emissions using a selected electricity emission

Copyright to IJARSCT
www.ijarsct.co.in

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

factor. Since the project does not use physical energy sensors, these values will be estimates rather than direct electricity measurements.

The system will also calculate a Green Score to present an indicative view of the application's resource efficiency. A Rule-Based Artificial Intelligence component will examine monitored conditions and provide recommendations according to predefined rules. For example, when CPU utilization remains high, the system may suggest reviewing CPU-intensive operations. When network activity is excessive, it may recommend reducing unnecessary requests or considering suitable caching methods. These recommendations will be based on the conditions and rules defined during system development. The project will use Python and Flask for the backend and HTML, CSS, and JavaScript for the web interface. A dashboard will display resource usage, estimated energy consumption, estimated CO_2 emissions, the Green Score, and the recommendations.

The main aim of this work is to develop a software-based system that helps developers observe resource usage and understand its estimated energy and carbon impact during web application execution. The proposed system will be implemented and tested to examine the behaviour of its monitoring functions, estimation method, Green Score, and rule-based recommendations.

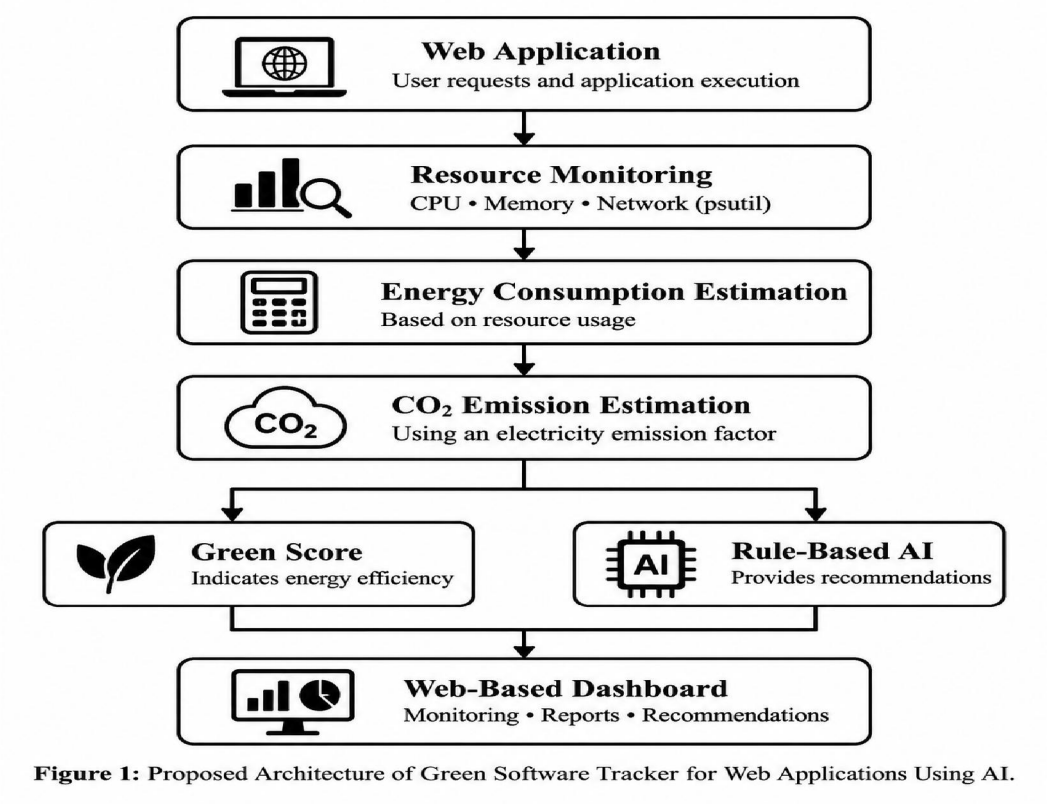


Figure 1: Proposed Architecture of Green Software Tracker for Web Applications Using AI.

II. METHODOLOGY

The proposed research focuses on developing a software-based Green Software Tracker for Web Applications Using AI. The system is designed to monitor the computational resource usage of a web application during its execution and estimate its energy consumption and associated carbon dioxide (CO_2) emissions. The proposed system will operate through a web-based interface and will not require additional physical energy monitoring hardware. It will collect system resource information from the computer running the application.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

The methodology begins with monitoring the resource usage of the target web application process. Python and the psutil library will be used to identify and monitor the selected process. The system will collect process-level CPU utilization, memory consumption, and available network activity information related to the application. This approach will help distinguish the resource usage of the target web application from the overall resource usage of the computer. The collected information will be processed by the proposed energy estimation method to calculate estimated energy consumption. The system will then use an appropriate electricity emission factor to estimate the associated CO_2 emissions.

After estimating energy consumption, the system will calculate the associated CO_2 emissions using an appropriate electricity emission factor. The estimated resource information will also be used to calculate a Green Score. This score will provide an indicative representation of the application's resource efficiency according to the scoring method defined for the proposed system.

The system will include a rule-based Artificial Intelligence component for generating optimization recommendations. It will examine the monitored resource usage and compare the values with predefined conditions. For example, high CPU utilization may generate a recommendation to optimize processor-intensive operations. Similarly, excessive network activity may generate suggestions to reduce unnecessary requests or use suitable caching techniques. The recommendation component will use predefined rules and will not require a Machine Learning model.

Python and Flask will be used to develop the backend functionality. HTML, CSS, and JavaScript will be used to develop the user interface. The dashboard will display CPU utilization, memory consumption, network activity, estimated energy consumption, CO_2 emissions, Green Score, and AI-generated recommendations. The system will also provide report generation functionality to present the monitoring information.

The proposed system will be evaluated through implementation and testing. Testing will examine resource monitoring, energy and carbon emission estimation, Green Score calculation, recommendation generation, and dashboard functionality. The results will help assess whether the proposed system performs its intended functions correctly.

A. System Architecture

The proposed system consists of four main components:

1. Data Collection Module: Identifies the selected web application process and collects CPU utilization and memory consumption at defined time intervals using Python and psutil.

2. Energy Estimation Module: Uses collected resource utilization data and a defined estimation method to calculate the estimated energy consumption of the application. The estimation method will require suitable assumptions and calibration before its accuracy can be established.

3. Carbon Emission Calculation Module: Calculates estimated carbon emissions using the estimated energy consumption and an appropriate electricity emission factor.

4. Dashboard and Recommendation Module: Displays resource utilization, estimated energy consumption, carbon emissions, Green Score, and suggestions for improving application efficiency.

B. Resource Monitoring

The system uses the Python psutil library to collect information about the selected web application process. It monitors the following parameters:

- **CPU Utilization:** Measures the percentage of CPU resources used by the selected application process.
- **Memory Consumption:** Records the amount of memory occupied by the application process.
- **Network Usage:** Monitors network activity where application-level traffic can be reliably identified.

The system collects these parameters at regular intervals. Process-level monitoring helps identify the resources consumed by the selected application. System-wide network statistics alone cannot accurately represent the network usage of an individual application.

C. Energy Consumption Estimation

The proposed system estimates energy consumption based on the monitored resource utilization of the application. CPU and memory usage provide information about application resource demand.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

The estimation process requires a defined relationship between resource utilization and energy consumption. The model must specify its assumptions, measurement interval, hardware configuration, and calibration method.

Since the proposed system does not currently use a physical power measurement sensor, its energy values will be treated as estimates rather than direct measurements.

D. Carbon Emission Calculation

The system calculates estimated carbon emissions using the estimated energy consumption and an electricity emission factor.

The general calculation is:

$$CO_2e = E \times EF$$

Where:

- CO_2e = Estimated carbon emissions
- E= Estimated energy consumption in kWh
- EF= Electricity emission factor in kg CO_2e/kWh

The emission factor must come from a reliable source and should match the relevant geographical region and reporting period.

E. Green Score Calculation

The proposed system includes a Green Score to represent the resource efficiency of the monitored web application.

The score will consider parameters such as CPU utilization, memory consumption, and estimated energy consumption.

A normalization method and weighting scheme will be defined to calculate the score on a scale of 0 to 100.

A higher score will represent better resource efficiency under the selected scoring criteria. The scoring formula will require validation before the score can be used for comparisons between applications.

F. Recommendation Module

The recommendation module analyzes the collected resource utilization data and provides suggestions based on predefined rules.

Examples include:

- If CPU utilization remains high for a sustained period, the system may suggest reviewing unnecessary background tasks or optimizing processing operations.
- If memory consumption increases continuously, the system may suggest checking memory leaks and optimizing memory allocation.
- If network usage is high, the system may suggest reducing unnecessary requests and optimizing data transfer.

The current recommendation approach is rule-based. Machine learning integration can be considered in future development.

G. Technology Stack

Component	Technology
Programming Language	Python
Backend Framework	Flask
Resource Monitoring	psutil
Frontend	HTML, CSS, JavaScript
Dashboard	Web-based dashboard
Data Analysis	Python

H. Proposed Workflow

The proposed system follows these steps:

1. Select the web application to monitor.
2. Identify its running process.
3. Collect CPU and memory utilization data.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

4. Analyze the collected resource information.
5. Estimate energy consumption using a defined method.
6. Calculate estimated carbon emissions.
7. Calculate the Green Score using predefined criteria.
8. Generate recommendations based on resource utilization.
9. Display the information through the web dashboard.

III. LITERATURE REVIEW

Previous studies on green computing and software energy consumption have examined different methods for understanding the environmental impact of software applications. Kalliola and Vepsäläinen [1] studied the challenges involved in estimating the energy consumption of websites. Their research examined different website energy estimation approaches and discussed CPU-based measurement using tools such as Firefox Profiler. The study provides useful information about estimating energy consumption during website execution. However, its main focus is energy measurement and estimation rather than providing an integrated system with a Green Score and automated optimization recommendations.

Paul et al. [2] presented a comprehensive review of green computing research covering past developments, current practices, and future research directions. Their study examined various green computing techniques, energy-efficient computing approaches, and methods for reducing environmental impact. It also discussed challenges and research opportunities in this field. However, the study mainly reviews existing research and does not implement a dedicated software tool for monitoring the resource consumption of web applications.

Pazienza et al. [3] discussed a holistic approach to environmentally sustainable computing through an Environmentally Sustainable Computing framework. Their work considered sustainability practices related to software design, development, monitoring, and improvement. It also discussed the application of sustainable computing practices in different computing environments. However, the framework addresses sustainability at a broader level and does not specifically provide the lightweight web application monitoring and recommendation system proposed in this research.

Based on the study of these existing works, the proposed Green Software Tracker for Web Applications Using AI focuses on monitoring CPU, memory, and network utilization through software-based techniques. It estimates energy consumption and associated CO_2 emissions and calculates a Green Score to represent the estimated resource efficiency of a web application. The system also uses a rule-based Artificial Intelligence component to examine resource usage conditions and provide suitable optimization recommendations. A web-based dashboard will present the monitoring information and generated reports in a single interface.

IV. RESULTS AND DISCUSSIONS

The proposed Green Software Tracker is designed to monitor the resource utilization of web applications and estimate their environmental impact. The system is expected to display CPU utilization, memory consumption, estimated energy consumption, carbon emissions, and Green Score through a web-based dashboard.

The collected resource data can help developers identify applications with high CPU or memory usage. The recommendation module is designed to provide suggestions for improving resource efficiency based on predefined rules.

The proposed framework has some limitations. Energy consumption is estimated without direct hardware power measurements. The accuracy of carbon emission calculations depends on the estimation method and the selected electricity emission factor. The Green Score also requires testing and validation.

IV. CONCLUSION

This paper proposes a software-based Green Software Tracker for Web Applications Using AI to estimate the energy consumption and associated carbon emissions of web applications. The proposed system monitors process-level CPU

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

utilization, memory consumption, and available network activity using Python and the psutil library. It uses the collected resource information to estimate energy consumption and calculate the corresponding CO_2 emissions using an appropriate electricity emission factor. The system also includes a Green Score to indicate the estimated resource efficiency of the application. A rule-based Artificial Intelligence component will examine resource usage conditions and provide recommendations for improving application efficiency. The web-based dashboard will present monitoring information, estimated energy consumption, carbon emissions, Green Score, and recommendations. The proposed work aims to help developers understand the resource requirements of their web applications and consider energy efficiency during software development. Future work may include improving the estimation method, evaluating the accuracy of the calculated values, and extending the monitoring features.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to their project guide for valuable guidance and support throughout this research work. The authors also thank the Department of Computer Engineering and Jaihind College of Engineering, Kuran, for providing academic support and necessary resources.

REFERENCES

1. Guldner et al., “Development and Evaluation of a Reference Measurement Model for Assessing the Resource and Energy Efficiency of Software Products and Components: Green Software Measurement Model (GSMM),” Future Generation Computer Systems, vol. 155, pp. 402–418, 2024.
2. J. Kalliola and J. Vepsäläinen, “Challenges Related to Approximating the Energy Consumption of a Website,” IEEE Access, vol. 13, pp. 139001–139017, 2025.
3. S. G. Paul et al., “A Comprehensive Review of Green Computing: Past, Present, and Future Research,” IEEE Access, vol. 11, pp. 87445–87494, 2023.
4. Pazienza et al., “A Holistic Approach to Environmentally Sustainable Computing,” Innovations in Systems and Software Engineering, vol. 20, pp. 347–371, 2024.