# Towards Sustainable Automation in Data Centers Using DevOps Methodologies

**Josiany B. Guimarães[1], Fellipe S. Queiroz[1], Erick D. Silva[1], Daniel Cordeiro[2], Marcos Amaris[1]**

[1]Universidade Federal do Pará, PPCA-NDAE - Tucuruí, PA

[2]Universidade de São Paulo, EACH - São Paulo, SP

{josiany.guimaraes, fellipe.queiroz, erick.silva}@tucurui.ufpa.br, daniel.cordeiro@usp.br, amaris@ufpa.br

***Abstract.*** *The increasing demand for digital services has intensified the expansion of data centers and their energy consumption. Infrastructure automation based on DevOps practices has emerged as a strategy to improve operational efficiency. This study investigates the impact of automation on energy consumption and carbon emissions in web server environments. An experimental evaluation compared the manual and automated execution of infrastructure configuration tasks, measuring execution time, energy consumption, and estimated $CO_2$ emissions. The results show that automation reduced execution time and energy consumption, leading to a 24.3% reduction in estimated carbon emissions and indicating its potential to support more energy-efficient computing environments.*

***Resumo.*** *A crescente demanda por serviços digitais intensificou a expansão dos data centers e o seu consumo de energia. A automação de infraestrutura, baseada em práticas DevOps, emergiu como uma estratégia para melhorar a eficiência operacional. Este estudo investiga o impacto da automação no consumo de energia e nas emissões de carbono em ambientes de servidores web. Uma avaliação experimental comparou a execução manual e a automatizada de tarefas de configuração de infraestrutura, mensurando o tempo de execução, o consumo de energia e as emissões estimadas de $CO_2$. Os resultados demonstram que a automação reduziu o tempo de execução e o consumo de energia, resultando em uma redução de 24,3% nas emissões estimadas de carbono e indicando seu potencial para viabilizar ambientes computacionais mais eficientes em termos energéticos.*

## 1. Introduction

The growing demand for digital services has driven the expansion of data centers, resulting in a significant increase in energy consumption associated with information technology infrastructure. In Brazil, recent data from the Ministry of Mines and Energy indicate an increase in requests to connect *data centers* to the national power grid, highlighting the strain these facilities place on the energy system [Ministério de Minas e Energia 2025]. In this context, energy efficiency becomes a fundamental aspect for the operational sustainability of these environments.

Several approaches have been proposed to mitigate the environmental impacts of IT infrastructure, notably the field of *Green Computing*, which seeks to promote the efficient use of computational resources and reduce emissions associated with energy consumption [Onoja et al. 2024]. Among the technological strategies adopted in this context, the *DevOps* paradigm has received growing attention for enabling greater integration between development and operations and for facilitating the automation of infrastructure management processes [Debois 2008].

Despite advances in automation and sustainable IT practices in IT environments, the literature still lacks empirical assessments of their impact on data center energy efficiency. In particular, experimental studies that quantitatively analyze the relationship between infrastructure automation, energy consumption, and $CO_2$ emissions remain limited, especially in the academic sphere [Maia Gomes et al. 2022].

Given this scenario, this study investigates whether adopting *DevOps* practices focused on infrastructure automation can reduce energy consumption and $CO_2$ emissions in *data centers* at higher education institutions. To this end, an experimental study was conducted to compare manual and automated execution of routine system administration tasks, evaluating metrics such as execution time, energy consumption, and estimated carbon emissions.

The remainder of this paper is organized as follows. Section 2 discusses the main concepts and theoretical background to understand this work. Section 3 presents the related works of this paper. Sections 4 describe the materials and methodology of this research. Section 5 presents the experimental results, comparative analyses, and discusses methodological limitations. Finally, Section 6 concludes the paper and mentions possible future works of this research.

## 2. Concepts and Background

The rapid growth of digital services, including cloud computing, streaming platforms, and large-scale web applications, has led to a substantial expansion of data center infrastructures. This expansion has been accompanied by a significant increase in energy consumption, making data centers a major contributor to global electricity demand in the information and communication technology (ICT) sector [Uddin and Rahman 2012, Reddy et al. 2018, International Energy Agency (IEA) 2023]. Consequently, improving energy efficiency and reducing environmental impact have become critical challenges in sustainable computing research.

In this context, adopting more efficient operational practices and technologies is central to mitigating the environmental footprint of computing infrastructure. Among these approaches, Green Computing and infrastructure automation have emerged as key strategies for enhancing both performance and sustainability [Onoja et al. 2024, Marco 2024].

### 2.1. Green Computing and Energy Efficiency

Green Computing has gained increasing attention in the context of sustainable IT as a way to address the environmental implications of computing infrastructures. In practice, it involves adopting techniques that reduce energy consumption, improve resource efficiency, and limit emissions associated with the operation of digital systems [Uddin and Rahman 2012]. More broadly, Green IT encompasses strategies to improve energy efficiency, optimize the use of computational resources, minimize waste, and reduce emissions associated with IT operations.

In data center environments, Green Computing strategies typically include dynamic resource allocation, server consolidation through virtualization, and the adoption of energy-efficient hardware. Enhancements in cooling technologies, such as hot- and cold-aisle containment and liquid-cooling solutions, also significantly reduce power usage effectiveness (PUE). In addition, workload optimization techniques aim to align computational demand with available resources, thereby minimizing overprovisioning and unnecessary energy consumption [Marco 2024, Uddin and Rahman 2012].

Beyond technical optimizations, energy efficiency in data centers is closely linked to environmental sustainability because of the carbon intensity of electricity generation. Since a significant portion of global energy production still depends on fossil fuels, reductions in energy consumption directly contribute to lowering carbon dioxide ($CO_2$) emissions [National Oceanic and Atmospheric Administration (NOAA) 2022]. Consequently, strategies that improve energy efficiency not only reduce operational costs but also support broader efforts to mitigate climate change.

### 2.2. Infrastructure Automation

Infrastructure automation has become a fundamental practice in modern IT environments, enabling the management, configuration, and provisioning of computing resources through software-driven processes rather than manual intervention [RedHat 2018, Reddy et al. 2018]. In this context, operational tasks such as system provisioning, software deployment, and environment configuration are executed using scripts and automation frameworks, ensuring greater consistency, repeatability, and standardization across infrastructure environments.

From an operational perspective, automation reduces the likelihood of human error, increases deployment speed, and improves system reliability. These characteristics are particularly relevant in large-scale infrastructures, where manual management becomes inefficient and prone to inconsistencies. Furthermore, automation aligns closely with DevOps practices, which emphasize collaboration between development and operations, as well as Continuous Integration and Continuous Delivery/Deployment (CI/CD) processes [Redhat 2022, Beck et al. 2001].

Beyond operational efficiency, infrastructure automation can also indirectly improve energy efficiency. Automated processes tend to execute tasks in a more controlled and predictable manner, reducing execution time and limiting the duration during which computational resources remain actively utilized. In addition, automation enables more efficient resource allocation and deallocation, helping to minimize idle or underutilized systems and, consequently, unnecessary energy consumption [Uddin and Rahman 2012, Reddy et al. 2018].

In this work, infrastructure automation is analyzed as a mechanism to evaluate how operational practices influence execution time, energy consumption, and associated $CO_2$ emissions in server environments. By comparing automated and manual execution approaches, this study provides insights into the potential of automation-driven practices to improve both operational efficiency and environmental sustainability in data center infrastructures.

## 3. Related Works

Research at the intersection of DevOps practices, infrastructure automation, and computational sustainability has attracted growing attention, particularly studies examining how operational practices influence energy consumption and carbon emissions in IT environments. In this context, recent work has explored integrating environmental metrics into the software development and operations lifecycle, as well as the energy implications of automated processes in data center infrastructure.

Ailane et al. [Ailane et al. 2025] propose a framework that integrates sustainability principles into the DevOps lifecycle, incorporating environmental indicators such as Software Carbon Intensity (SCI) and energy consumption metrics across planning, implementation, and operation. The study highlights the role of automation in collecting and monitoring environmental metrics within continuous integration and delivery pipelines. Similarly, Muppala et al. [Muppala 2025] analyze automation strategies to reduce data center carbon footprints through infrastructure automation, containerization, and energy-monitoring mechanisms.

Other studies investigate techniques for predicting energy consumption in data centers. For example, works such as [Chatlatanagulchai and Chantrapornchai 2024] and [Zhang and Liu 2022] employ machine learning approaches to analyze operational data and model energy consumption patterns, enabling more efficient resource management and improved infrastructure utilization.

In the Brazilian context, Lunardi et al. [Lunardi et al. 2014] discuss Green IT practices, including server virtualization, energy-efficient use, and environmentally responsible disposal of electronic equipment.

Overall, the studies discussed in this section have proposed methods and frameworks to improve sustainability in computing infrastructures, although important methodological differences can be observed among them. Some studies adopt framework-oriented approaches based on integrating environmental metrics into CI/CD pipelines, while others employ predictive machine learning models, containerization strategies, or operational energy-monitoring mechanisms in cloud and data center infrastructures.

Despite these differences, the studies share the common objective of improving operational efficiency and reducing the environmental impact of computing environments. The diversity of approaches also suggests that there is still no consolidated methodological standard for integrating sustainability practices into infrastructure management workflows, with each study emphasizing different optimization strategies and operational priorities.

In this context, infrastructure automation emerges as one of the operational strategies capable of contributing to sustainability initiatives by reducing repetitive manual interventions, optimizing resource utilization, and potentially minimizing unnecessary energy consumption in computing environments.

# 4. Methodology

## 4.1. Experimental Environment

The experiments were conducted in the institutional data center of a public university. The infrastructure is hosted in a containerized data center equipped with precision cooling, a grid-connected power supply with battery backup systems, and network connectivity. The virtualized environment comprises multiple servers that host academic and administrative services, including web applications, databases, authentication systems, and monitoring tools.

For the experiments, a simplified environment was defined consisting of one control node and two target virtual machines. The control node, running Debian 12, orchestrated the experiments, while the target nodes were typical infrastructure servers used in institutional environments.

## 4.2. Automation Workflow

Infrastructure automation was implemented through automation scripts executed from the control node. Each automated task reproduces routine system administration operations and follows the same logical steps as the corresponding manual execution.

This design ensures comparability between manual and automated approaches, enabling controlled evaluation of execution time, energy consumption, and associated carbon emissions.

## 4.3. Energy Monitoring and Estimation

Energy consumption data were collected using the Integrated Monitoring and Management System (IMAMS), which monitors electrical parameters of the data center, including accumulated energy consumption.

Because the monitoring system provides measurements at fixed intervals rather than at each task execution, a proportional estimation approach was adopted. The energy consumption associated with each task was calculated based on the ratio between the task execution time and the monitoring interval, as follows:

$$E = \left( \frac{t_{\text{exec}}}{T_{\text{interval}}} \right) \times E_{\text{total}} \tag{1}$$

where $E$ represents the estimated energy consumption of the task, $t_{\text{exec}}$ is the execution time, $T_{\text{interval}}$ is the monitoring interval, and $E_{\text{total}}$ is the total energy recorded in that interval.

This approach provides a consistent and practical estimate of task-level energy consumption in real-world environments where fine-grained measurements are unavailable. However, it assumes a uniform distribution of energy consumption within the monitoring interval, which may introduce minor inaccuracies for short-duration tasks.

## 4.4. Evaluated Tasks

The evaluated tasks correspond to routine infrastructure management activities typically performed by system administrators. These include database backup and restoration, web server deployment, security policy configuration, and application update procedures.

Each task was performed in two modes: manual and automated. All experiments were conducted under equivalent conditions to ensure comparability.

### 4.5. $CO_2$ Emission Estimation Based on Measured Energy Consumption

$CO_2$ emissions were estimated based on the measured energy consumption of each task, using the Green Algorithms model [Lannelongue et al. 2021]. This approach establishes a direct relationship between electricity consumption and carbon emissions via the energy source's emission factor.

In this study, the emission factor of the Brazilian electricity matrix was adopted, defined as 54.5 $gCO_2$/kWh, based on official data from the national energy system.

The estimation of carbon emissions was performed according to Equation 2:

$$CO_2 = E \times 54.5 \quad (2)$$

where $E$ represents the energy consumption in kWh and the resulting emissions are expressed in grams of $CO_2$.

This methodology enables a consistent and reproducible estimation of the environmental impact associated with each evaluated task, allowing direct comparison between manual and automated executions in terms of carbon emissions.

### 4.6. Experimental Design

The experimental evaluation was designed to compare manual and automated execution under equivalent conditions, ensuring consistency across all experiments.

Each task was executed independently in both modes, using the same input data and system configuration. The evaluation focused on three metrics: execution time, estimated energy consumption, and associated $CO_2$ emissions.

This design enables the isolation of the impact of automation on performance and energy efficiency while minimizing external interference from environmental factors. Additionally, this methodology bridges the gap between coarse-grained energy monitoring and task-level energy analysis in real-world data center environments.

### 4.7. Use of Writing Assistance Tools

Artificial intelligence-based writing assistance tools, including *ChatGPT* [OpenAI 2025] and *Grammarly* [Grammarly Inc. 2025], were used exclusively to support linguistic revision and improve textual clarity. The authors bear full responsibility for the conceptual, methodological, and analytical content of this work.

## 5. Results and Discussion

This study presents experimental results from the execution of infrastructure management tasks in both manual and automated modes. The analysis focuses on execution time, estimated energy consumption, and associated $CO_2$ emissions.

To ensure reproducibility, all automation scripts, configuration files, and supporting materials used in this study are publicly available in a GitHub repository: `https://github.com/josianybrito/ansible`.

### 5.1. Overview of Experimental Results

The evaluated tasks include database backup and restoration, web server deployment, security configuration, and application update procedures. Each task was executed under equivalent conditions in both manual and automated modes.

Table 1 presents a consolidated comparison of the experimental results, including execution time, estimated energy consumption, calculated $CO_2$ emissions, and the corresponding percentage of energy savings achieved through automation.

Overall, the results indicate a consistent improvement across all evaluated metrics. The total execution time decreased from 1240.66 seconds in manual execution to 949.97 seconds in automated execution. Similarly, total energy consumption was reduced from 0.41672 kWh to 0.31552 kWh, corresponding to an overall reduction of approximately 24.28%.

These findings demonstrate that automation contributes not only to improved operational efficiency but also to enhanced energy performance in infrastructure management tasks.

**Table 1. Comparison between manual and automated execution of evaluated tasks.**

| Task | T-Manual (s) | Energy-M (kWh) | T-Auto (s) | Energy-A (kWh) | Savings (%) | $CO_2$ Manual (g) | $CO_2$ Auto (g) |
|---|---|---|---|---|---|---|---|
| Web Server Deployment | 56.38 | 0.0137 | 43.68 | 0.0098 | 28.47 | 0.75 | 0.53 |
| Database Operations | 106.00 | 0.0260 | 107.00 | 0.0230 | 11.54 | 1.42 | 1.25 |
| Security Configuration | 4.28 | 0.00102 | 3.29 | 0.000719 | 29.51 | 0.06 | 0.04 |
| Application Update | 1074.00 | 0.3760 | 796.00 | 0.2820 | 25.00 | 20.49 | 15.37 |
| **Total** | **1240.66** | **0.41672** | **949.97** | **0.31552** | **24.28** | **22.72** | **17.19** |

As shown in Table 1, automation consistently reduces both execution time and energy consumption across all evaluated tasks.

### 5.2. Energy Consumption Analysis

Automation consistently reduced energy consumption across all evaluated tasks. The observed reductions range from approximately 11.54% in database operations to 29.51% in security configuration tasks.

Tasks involving multiple sequential operations, such as web server deployment and security configuration, exhibited the highest energy savings. This behavior can be attributed to reduced idle time, fewer execution delays, and more efficient orchestration of commands enabled by automation.

In contrast, database operations yielded smaller gains because they depend on I/O-bound processes, for which execution efficiency is inherently less sensitive to automation.

These results indicate that energy savings are strongly influenced by task complexity and execution structure.

### 5.3. Execution Time Analysis

Automation reduced execution time in most evaluated scenarios. The web server deployment task showed a reduction of approximately 22.5%, while the application update task presented a reduction of approximately 25.8%.

Although database operations exhibited minimal variation in execution time, automation ensures greater consistency and reduces the likelihood of human-induced delays or errors.

From this perspective, the results suggest a direct relationship between reductions in execution time and improved energy efficiency, particularly for tasks with longer execution times.

### 5.4. $CO_2$ Emission Analysis

Based on the estimated energy consumption, $CO_2$ emissions were calculated using the emission factor of the Brazilian electricity matrix (54.5 $gCO_2$/kWh).

The total emissions decreased from 22.72 $gCO_2$ in manual execution to 17.19 $gCO_2$ in automated execution, representing a reduction of approximately 24.3%.

Although the absolute emission values per task are relatively small, their cumulative impact becomes significant in environments where such tasks are executed repeatedly.

### 5.5. Detailed Comparative Analysis

A more detailed analysis reveals a strong relationship between execution time, energy consumption, and carbon emissions. Tasks with longer execution times tend to consume more energy and consequently emit more.

Automation reduces execution overhead by minimizing idle periods and ensuring continuous task execution, thereby improving CPU utilization.

The highest relative energy savings were observed in security configuration tasks (29.51%), likely due to the sequential and repetitive nature of the operations. In contrast, database operations showed smaller improvements because they are constrained by I/O performance.

These observations indicate that the effectiveness of automation in reducing energy consumption is closely related to the task's structure and complexity.

### 5.6. Methodological Limitations

The methodology adopted in this study uses a proportional estimation approach to infer task-level energy consumption from coarse-grained monitoring data. While practical, this approach introduces certain limitations.

1. First, the IMAMS system provides measurements at fixed intervals, requiring the assumption of a uniform distribution of energy consumption within each interval. This assumption may introduce approximation errors, particularly for short-duration tasks.
2. Second, background processes and concurrent workloads in the data center may influence the measured energy consumption, introducing potential noise into the estimation.
3. Third, the methodology does not capture fine-grained hardware-level variations, such as CPU frequency scaling or memory access patterns, which may affect energy consumption.

4. Fourth, the experiments were conducted in a specific academic data center environment using representative administrative tasks, which may limit the generalization of the results to other infrastructure scenarios, workload profiles, or production environments.
5. Finally, the estimated $CO_2$ emissions were calculated using a fixed emission factor based on the Brazilian electrical matrix, providing a simplified approximation of the environmental impact associated with energy consumption.

Despite these limitations, the proposed approach provides a consistent and reproducible framework for experimental and comparative analysis, enabling the evaluation of different infrastructure management approaches according to methodologies commonly adopted in empirical software engineering and software energy-consumption studies [Travassos et al. 2002, Hindle 2015].

### 5.7. Implications for Performance and Energy Efficiency

The results demonstrate that infrastructure automation directly impacts both performance and energy efficiency.

From a performance perspective, automation reduces execution time by minimizing manual intervention and ensuring consistent execution paths. From an energy perspective, reduced execution time leads to lower energy consumption, establishing a direct relationship between performance optimization and energy efficiency.

This relationship highlights the importance of incorporating energy-aware considerations into performance engineering practices, particularly in data center environments.

### 5.8. Implications for Large-Scale Environments

Although per-task energy savings are relatively small in absolute terms, their cumulative impact becomes significant in large-scale environments.

In data centers, infrastructure management tasks are executed frequently across multiple systems. Therefore, even modest efficiency improvements can result in substantial reductions in overall energy consumption and carbon emissions.

This reinforces automation as a scalable strategy for improving the sustainability of IT operations.

### 5.9. Comparative Analysis of Carbon Estimation Models

This study adopts the Green Algorithms model [Lannelongue et al. 2021], which provides a practical, widely accepted approach for estimating carbon emissions from measured energy consumption.

In contrast, simulation-based approaches such as SimGrid/Batsim enable fine-grained and dynamic estimation of energy consumption over time [Saraiva et al. 2025].

The choice of the Green Algorithms model is justified by its compatibility with empirical data and its suitability for environments with only coarse-grained measurements.

## 6. Conclusions and Future Works

This study investigated the impact of infrastructure automation based on DevOps practices on the operational and energy efficiency of web server environments. An experimental evaluation was conducted to compare manual and automated execution of routine system administration tasks, using metrics including execution time, energy consumption, and estimated $CO_2$ emissions.

The results indicate that infrastructure automation can improve operational efficiency while reducing environmental impact. The automated approach reduced task execution time and energy consumption, resulting in an average 24.3% reduction in estimated carbon emissions compared to the manual configuration process.

These findings suggest that adopting DevOps-oriented automation can not only improve the reproducibility and standardization of infrastructure management but also promote more energy-efficient and environmentally sustainable computing environments.

Although improvements in operational efficiency through automation may be expected from a systems administration perspective, this study contributes by providing an empirical and quantitative evaluation of the relationship between infrastructure automation, execution time, energy consumption, and estimated $CO_2$ emissions in a real academic infrastructure environment.

It is important to note that the experiments were conducted within a real academic production data center environment using a dedicated test server, which imposes practical constraints on the execution of large-scale or stress-intensive workloads. To preserve service availability and avoid potential disruptions to end users, the evaluated tasks were limited to representative administrative operations. Although this approach reflects realistic operational conditions, it may restrict the generalization of the results to other workload types.

Future work may expand this investigation by analyzing energy efficiency across broader DevOps workflows, including continuous integration, orchestration, and container management platforms. In addition, further studies could evaluate the impact of automation in different infrastructure scenarios, incorporate a larger number of repeated executions and statistical analyzes, including variability and dispersion metrics such as standard deviation and confidence intervals, and explore more diverse workload profiles, thereby contributing to a deeper understanding of the relationship between infrastructure automation and sustainable computing practices.

## Acknowledgements

This work was partially supported by the National Council for Scientific and Technological Development (CNPq, grant no. 404352/2024-3), by the São Paulo Research Foundation (FAPESP, grants no. 19/26702-8 and 23/00811-0), and by the Science for Development Center "Carbon-Neutral Cities" (FAPESP grant no. 24/01115-0).

## References

Ailane, M. T., Rubner, C., and Rausch, A. (2025). Green devops: A strategic framework for sustainable software development. In *Computer Sciences & Mathematics Forum*, volume 10, page 5. MDPI. Article 5.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., et al. (2001). Manifesto for agile software development. `https://agilemanifesto.org/`.

Chatlatanagulchai, W. and Chantrapornchai, C. (2024). Energy consumption data collection: case study on data center in a thai university. *Energy Informatics*, 7(1):26.

Debois, P. (2008). Agile infrastructure and operations: How infra-gile are you? In *Agile 2008 Conference*, pages 202–207. IEEE.

Grammarly Inc. (2025). Grammarly: AI writing assistance.

Hindle, A. (2015). Green mining: a methodology of relating software change and configuration to power consumption. *Empirical Software Engineering*, 20(2):374–409.

International Energy Agency (IEA) (2023). Data centres and data transmission networks. Available at: https://www.iea.org/reports/data-centres-and-data-transmission-networks.

Lannelongue, L., Grealey, J., and Inouye, M. (2021). Green algorithms: quantifying the carbon footprint of computation. *Advanced science*, 8(12):2100707.

Lunardi, G. L., Simões, R., and Frio, R. S. (2014). TI verde: Uma análise dos principais benefícios e práticas utilizadas pelas organizações. *REAd. Revista Eletrônica de Administração (Porto Alegre)*, 20:1–30.

Maia Gomes, K. G., de Lima Albuquerque, J., Cabral, R. M., Benevides de Pinho, M. A., and da Silva Correia-Neto, J. (2022). Práticas de tecnologia da informação verde em instituições de ensino superior: Uma revisão sistemática. *Id on Line. Revista de Psicologia*, 16(60).

Marco, K. (2024). Soluções de TI e eficiência energética para data centers. In *Tecnologia, Comunicações e Gestão da Informação: Contribuições para o Desenvolvimento de Novos Conhecimentos*, volume 2, pages 185–197. Editora Científica Digital.

Ministério de Minas e Energia (2025). MME identifica crescimento acelerado na demanda de energia elétrica para projetos de data centers. Acesso em: 13 fev. 2025.

Muppala, P. K. (2025). Sustainable devops: Minimizing the carbon footprint of banking data centers. *International Journal of Science and Research Archive*, 14(1):1780–1793.

National Oceanic and Atmospheric Administration (NOAA) (2022). Carbon toolkit - basics. `https://gml.noaa.gov/education/carbon_toolkit/basics.html`. Acesso em: 24 jul. 2023.

Onoja, M. O., Onyenze, C. C., and Akintoye, A. A. (2024). Devops and sustainable software engineering: Bridging speed, reliability, and environmental responsibility. *International Journal of Technology Management & Humanities*.

OpenAI (2025). ChatGPT: ferramenta de inteligência artificial. Disponível em: `https://chat.openai.com/`. Acesso em: 6 dez. 2025.

Reddy, V. D., Setz, B., G. Subrahmanya V. R. K. Rao, G., Gangadharan, G. R., and Aiello, M. (2018). Best practices for sustainable datacenters. *IT Professional*, 20(5):57–67.

RedHat (2018). O que é a automação de TI? `https://www.redhat.com/pt-br/topics/automation/whats-it-automation`. Acessado em 28/12/2023.

Redhat (2022). Introdução ao devops. `https://www.redhat.com/pt-br/topics/devops`. Acesso em: 10 jun. 2023.

Saraiva, G., Vasconcelos, M., Bruschi, S. M., Carastan-Santos, D., and Cordeiro, D. (2025). Estimating $CO_2$ emissions of distributed applications and platforms with SimGrid/Batsim. In *Simpósio em Sistemas Computacionais de Alto Desempenho (SSCAD)*, pages 109–120. SBC.

Travassos, G. H., Gurov, D., and Amaral, E. (2002). *Introdução à engenharia de software experimental*, volume 9. UFRJ Rio de Janeiro, Brazil.

Uddin, M. and Rahman, A. A. (2012). Energy efficiency and low carbon enabler green it framework for data centers considering green metrics. *Renewable and Sustainable Energy Reviews*, 16(6):4078–4094.

Zhang, Y. and Liu, J. (2022). Prediction of overall energy consumption of data centers in different locations. *Sensors*, 22(10).