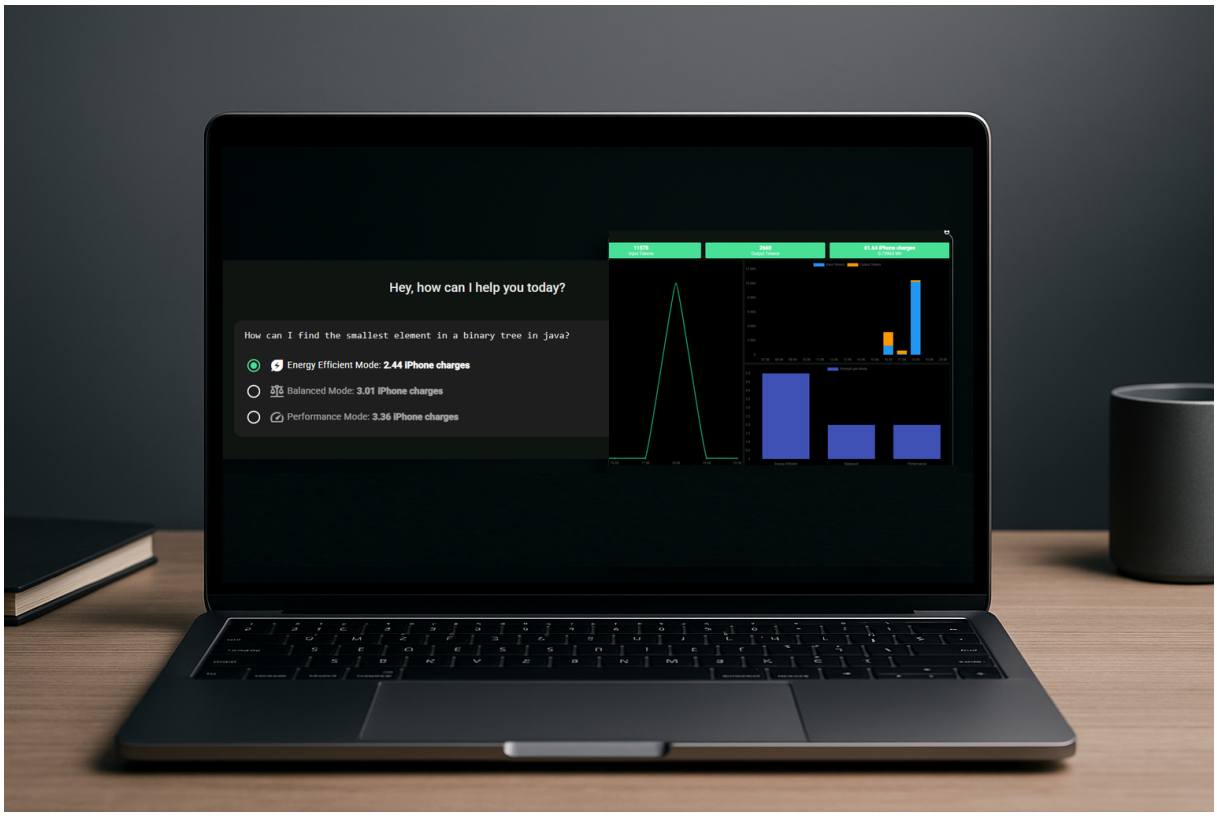
Fachhochschule Nordwestschweiz
Hochschule für Technik

UI Strategies for Reducing Conversational AI Energy Consumption

Jack Gläser and Simon Lüscher

Windisch, August 2025



Students	Jack Gläser and Simon Lüscher
Expert	Romano Roth
Supervisors	Prof. Martin Kropp and Dr. Nitish Patkar
Stakeholder	Fachhochschule Nordwestschweiz - Hochschule für Technik
Project number	IMVS-24

Fachhochschule Nordwestschweiz, Hochschule für Technik

Abstract

The rapid adoption of large language models (LLMs) for everyday information-seeking is shifting energy demand from highly optimized search engines to far more power-hungry conversational AI systems [1], [2]. Yet this cost remains largely invisible to end users.

This thesis investigates whether user interface (UI)-only interventions can increase user awareness and lead to measurable reductions in the energy consumption associated with chatbot usage.

We first conducted a baseline survey (n = 50), which revealed both awareness deficits and strong support for transparency features such as energy indicators and low-power modes. Guided by these findings and prior work in human computer interaction (HCI), we developed a full-stack ChatGPT-style prototype ("The Botter") incorporating five sustainability-focused UI features: (1) a three-mode toggle, (2) prompt-level energy prediction, (3) per-response Energy-Notes, (4) a usage metrics dashboard and (5) personalized energy analogies.

In a five-day field experiment involving eleven frequent LLM users, we observed a substantial increase in energy awareness (peaking at M = 4.44 on a 5 point Likert scale), alongside high usability ratings (all features above 4/5). Importantly, higher awareness correlated with more energy-efficient usage behavior: Participants in the top awareness quartile selected the energy-saving mode for 72% of their prompts, compared to 34% in the bottom quartile. Overall, more than half of all prompts were routed through the energy-efficient mode, yielding an estimated 35% reduction in energy consumption relative to a performance-mode baseline, without degrading user experience.

These findings demonstrate that lightweight, frontend design interventions can effectively nudge sustainable behavior in conversational AI interfaces. This contributes to addressing a key challenge in the design of human-AI interaction and highlights the role of interface design in mitigating the environmental impact of large-scale language model usage.

Acknowledgement

We would like to express our sincere gratitude to Prof. Martin Kropp and Dr. Nitish Patkar for their steady supervision, constructive feedback, and constant encouragement throughout this project.

We also thank Dr. Pooja Rani (University of Zurich) for her insightful advice on the experimental design and consent procedures, which markedly improved the study's quality.

Finally, a heartfelt thank-you to all who took part in our research: The 50+ survey respondents who provided the baseline insights and the eleven experiment participants who devoted a full week to testing the prototype. Your time and input made this thesis possible.

Thank you all.

Contents

List of Figures

List of Tables

Glossary

LLM Large Language Model. AI model trained on vast text data to generate human-like language, e.g., GPT-4.

UI User Interface. The means by which a user interacts with a computer system or software.

SPA Single-Page Application. A web application that loads a single HTML page and dynamically updates content.

CO_2 Carbon Dioxide. A greenhouse gas, often referenced in the context of emissions from computational processes.

Wh Watt-hour. A unit of energy used to quantify electricity consumption.

RAG Retrieval-Augmented Generation. An AI technique that combines information retrieval with generative models.

Azure Functions A serverless compute service provided by Microsoft Azure for running event-driven code.

Cosmos DB A globally distributed, multi-model database service by Microsoft Azure.

ChatGPT A conversational AI model developed by OpenAI, based on the GPT architecture.

Energy-Efficient Mode A system mode designed to minimize energy consumption, often by using smaller models or reduced computation.

Performance Mode A system mode prioritizing response quality and speed, typically using larger models and more computation.

Balanced Mode A compromise between energy efficiency and performance in system operation.

Token In NLP, a unit of text (word, subword, or character) processed by language models.

Inference The process of running a trained AI model to generate predictions or responses.

Prompt The input text provided to an AI model to elicit a response.

Dashboard A user interface element that displays key metrics and statistics, such as energy usage.

Awareness User knowledge of the energy and environmental impact of their AI usage.

HCI Human-Computer Interaction. The study and design of how people interact with computers and technology.

FLOPs Floating Point Operations per Second. A measure of computational performance, often used to describe the processing power required by AI models.

Heuristic A practical method or approach to problem-solving that is not guaranteed to be optimal but is sufficient for reaching an immediate goal.

Latency The delay between a user's action and the system's response, often measured in milliseconds.

Modalities Different forms or types of input/output data (e.g., text, image, audio) that an AI system can process.

Prompt Engineering The practice of designing and refining input prompts to optimize the output of language models.

Sustainability Meeting present needs without compromising the ability of future generations to meet their own needs, often referenced in the context of environmental impact.

Tokenization The process of breaking text into smaller units (tokens) for processing by language models.

Usability The ease with which users can effectively and efficiently interact with a system or product.

LOWESS Locally Weighted Scatterplot Smoothing. A non-parametric regression method that fits simple models to localized subsets of data to create a smooth curve through points in a scatterplot.

Empiric/Empirical Based on observation or experience rather than theory or pure logic; often refers to data or results derived from experiments or real-world evidence.

Regression A statistical method for modeling the relationship between a dependent variable and one or more independent variables.

Non-parametric Refers to statistical methods that do not assume a specific distribution for the data.

1 Introduction

1.1 Initial Situation

Conversational AI consumes a lot of energy during its entire life cycle [3]. A significant part of this energy is required for the development and training of the model [4], [5]. But the actual conversational AI service also consumes a great amount of energy during inference. Although a single prompt does not have a significant impact on the required energy, the cumulative effect grows rapidly with a large user base. In fact, inference energy usage now outweighs training in many large-scale deployments: If chatbot-based interfaces were to completely replace traditional web search, global electricity demand could increase by several tens of terawatt hours annually [3]. During one month, the inference energy can already exceed that of the model training [6]. Bond Capital reports that ChatGPT reached 800 million weekly active users just 17 months after launch, implying trillions of inference calls per month [7]. These figures underscore why inference energy has become a central environmental concern.

1.2 Problem Statement

Despite this growing footprint, most end users remain unaware of the hidden environmental costs of their interactions with large language models (LLMs). Traditional searches have well-optimized, transparent energy profiles, but LLM-based chat incurs substantial energy consumption. In what follows, we outline the key dimensions of this problem.

1.2.1 Perceptual Gap: Lack of User Awareness

Most users underestimate the energy implications of AI services. Our survey (Section 3.2) shows participants vastly misjudge per–query consumption, echoing Sustainable HCI findings that even tech-savvy users lack calibrated mental models for digital energy use [8]. Without awareness, users cannot factor sustainability into everyday choices.

1.2.2 Technical Gap: Barriers to Live Transparency

Accurate, per-query energy feedback is technically complex. Current estimation tools are not very accurate, offline, or not integrated into user interfaces [9], [10]. Live token-level attribution remains an open challenge behind proprietary, cloud-based APIs. Without precise, real-time estimates, transparency features risk being symbolic rather than informative.

1.2.3 Design Gap: Platform Incentives and UI Constraints

Commercial providers often deprioritize energy transparency when performance, engagement, and monetization are at stake. Defaults prioritize speed and fluency over efficiency, and although model-selection or prompt-optimization can cut inference energy by up to 60 % [11], [12], these options are rarely exposed to users. The design ecosystem lacks strong nudges toward sustainable behavior.

1.2.4 Problem Synthesis

Taken together, these facets reveal three interrelated gaps that this thesis addresses:

1. **Perceptual Gap** — Users are unaware of the true environmental cost of LLM usage.
2. **Technical Gap** — Live, accurate energy estimation is difficult to implement and deploy.
3. **Design Gap** — Current UIs and provider incentives offer little support for sustainability-aware decisions.

1.2.5 Opportunities for Intervention

Research in adjacent domains shows that real-time, contextual feedback can shift behavior, for example smart-meter dashboards and mobile energy apps have driven measurable reductions in consumption [13], [14]. UI-only strategies such as dashboards, mode toggles, and energy analogies can make backend energy innovations visible and actionable at the user level. This thesis builds on these insights to explore how purely interface-based interventions can raise awareness and nudge more sustainable use of conversational AI.

1.3 Overall Goal and Requirements

We strive to reduce the overall energy consumption associated with conversational AI by improving user awareness through targeted UI-based interventions. By researching and contributing to the state of the art, this thesis aims to understand how interface design alone can effectively influence user behavior regarding energy consumption. We will identify and evaluate UI features that increase user awareness, providing transparency and actionable insights into the energy impact of their interactions with conversational AI. Ultimately, our goal is to empower users to make informed decisions when and how to use conversational AI, fostering more sustainable and energy-efficient usage patterns without compromising the user experience.

1.3.1 Research Questions

Therefore, our work is guided by the following three research questions:

RQ1: *To what extent are users currently aware of the energy implications associated with their chatbot interactions?*

RQ2: *How can UI-based features most effectively increase user awareness regarding the energy consumption of conversational AI?*

RQ3: *How strongly does increased user awareness correlate with reductions in conversational AI energy consumption?*

1.4 Structure of the Thesis

This thesis is organized into eight core chapters, followed by a declaration of honesty and a comprehensive appendix.

- **Chapter 1 - Introduction** situates the study, delineates the problem space, formulates the research questions, and specifies the goals and requirements that guide the work.
- **Chapter 2 - Methodology** details the five-phase research design: Literature review, the baseline survey, prototype development, controlled experiment, and data-analysis plan.
- **Chapter 3 - State of the Art** reviews prior work on user awareness, energy footprints of conversational AI, UI-based sustainability interventions, and identifies the resulting knowledge gaps addressed in this thesis including our own survey results.
- **Chapter 4 - Conceptual Solution** outlines the high-level approach, describes the baseline chatbot and the five UI features (three-mode switch, metrics dashboard, prompt prediction, energy note, and energy analogies) and presents the underlying energy-estimation model.
- **Chapter 5 - Implementation** explains the concrete realisation of the system architecture on Azure, covering frontend, backend, data storage, non-functional requirements, and CI/CD pipeline.
- **Chapter 6 - Validation and Results** reports quantitative and qualitative findings from the user study(experiment), including awareness trajectories, behavioral metrics derived from server logs, and usability assessments.

- **Chapter 7 - Discussion** interprets the empirical results in light of the research questions and hypotheses, reflects on methodological and technical limitations, and outlines avenues for further research and development.
- **Chapter 8 - Conclusion** synthesises the key contributions and practical implications of the thesis.

2 Methodology

The following section outlines the five phases of the methodology that guided the development of this thesis. Each phase is briefly described to provide an overview of the overall approach.

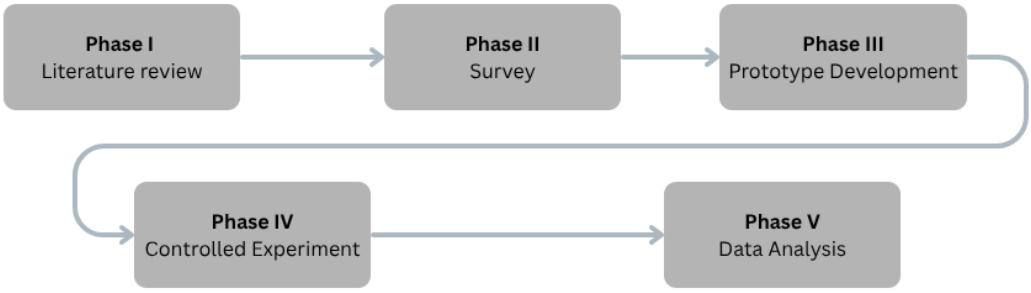


Figure 2.1: Overview of the five-phase thesis methodology

2.1 Phase I – Literature Review

We conducted a structured literature review to identify existing research and solutions related to sustainable AI and energy awareness in conversational systems. The process included:

- **Databases Searched:** Google Scholar, Google search and AI (ChatGPT deepsearch functionallity) for literature searches related to the subject.
- **Keywords Used:** Our search included combinations of terms such as “chatbot”, “LLM”, “large language models”, “sustainable AI”, “green AI”, “energy efficiency”, “carbon footprint AI”, “eco-feedback”, “energy”, “consumption”, “behavior change”, “energy literacy”, “gamification”, “peer comparison”, “eco-nudges”, “token efficiency”, “budget-constrained inference”, “carbon tracking widgets”, “HCI for sustainability”, and “sustainability metrics”. We adapted and expanded the keyword set iteratively as themes emerged.
- **Selection Criteria:** We prioritized peer-reviewed papers, highly cited publications, and industry reports published between 2016 and 2025. Additionally, we considered emerging tools (such as browser extensions) and preprints to gain insight into current trends and early-stage research on energy feedback and user-facing interventions in conversational AI.
- **Screening Process:** Abstracts and tool descriptions were screened for relevance to user awareness, energy consumption measurement, and UI-driven strategies for promoting sustainable behavior or raising awareness in chatbot interfaces or similar fields.

We totally collected 67 documents, articles or other tools and sources related to this keywords and used 45 directly in this thesis.

This methodology ensured that our review was both systematic and focused, providing a solid foundation for the analysis and proposed solutions presented in subsequent sections. The results of this review presented throughout the whole thesis directly shaped the structure of multiple sections such as the problem analysis and state-of-the-art, as well as the selection of features for the prototype development.

2.2 Phase II – Survey

To complement the literature review, we conducted a survey to assess the current user awareness and attitudes toward energy consumption in conversational AI. The survey methodology included:

- **Instrument:** The survey comprised fourteen closed items on a 5-point Likert scale [15] and three open questions.
- **Recruitment:** Participants were recruited via E-Mail, WhatsApp, LinkedIn, and in-person. Fifty valid responses were collected.
- **Data Collection:** Responses were collected anonymously through Google Forms, with an option for participants to add their email to volunteer for subsequent experiment.
- **Analysis Plan:** Descriptive statistics and correlation analysis was conducted to investigate the users awareness and examine their acceptance for energy consumption reducing UI-features.

The survey results are reported in Section 3.2 and informed the design of subsequent phases.

2.3 Phase III – Prototype Development

The development of the proof-of-concept chatbot (further called: *The Botter*) reproduces the ChatGPT interface and adds five sustainability modules which we wanted to test and research the effectiveness of:

a) **Energy note** below each response (Wh and user-chosen equivalent, e.g. "0.53 Wh which equals to 43 s laptop use").
b) **Prompt prediction** shows the estimated energy cost before sending a query.
c) **3-mode toggle** (Energy efficient, Balanced, Performance) with token-based cost prediction shown before sending to indicate that more powerful modes equal more energy consumption.
d) **Usage-metrics dashboard** aggregating per-day Wh, modes share and more.
e) **Energy-analogies** describe energy consumption in relatable units.

2.4 Phase IV – Controlled Experiment

2.4.1 Experimental Design

To evaluate the impact of our sustainability-focused UI features, we conducted a five-day controlled user study using our prototype (*The Botter*). The experiment was designed as a single-group field study, maximizing statistical power within the constraints of a small sample size [16]. All sustainability features were enabled by default, but participants could disable them at any time. System interactions, self-reports (daily check-ins and a final questionnaire at the end of the experiment) and background telemetry were logged for evaluation.

2.4.2 Hypotheses

This experiment was guided by the research questions defined in the section 1.3.1. Based on these questions, we formulated two hypotheses:

H1. Showing per-prompt consumption (*Energy-Note*) and predicted consumption (*Prompt Prediction* + *Three-Mode Switch*) increases the average awareness score from pre- to post-study.
H2. Individual awareness scores (end-of-study) are positively correlated with positive sustainability behavior logged during the study.

2.4.3 Participants and Duration

Eleven frequent LLM users (self-reported usage of $\geq$ 15 minutes per day in the baseline survey) were recruited from the survey pool. All participants live and work in Switzerland and demonstrated English proficiency, which is required because the prototype chatbot operates in English only. The sample spanned an age range of 25–35 years. Most participants were employed in IT-related roles or business services (e.g. software engineering, support, mortgage and financial consulting); one participant worked

in child care, providing some occupational diversity beyond the tech and finance sectors. All participants signed a consent document before starting the experiment, see A.2.1. The study ran across five consecutive workdays (Monday 23 June 2025 to Friday 27 June 2025). Each participant interacted with the prototype in their usual professional or academic context.

2.4.4 Instrumentation and Logged Metrics

Data collection combined three sources:

- Daily check-ins consisting of four Likert scale items on awareness and feature salience, see Appendix A.2.2
- A final questionnaire covering awareness, behavior, usability, and open-ended feedback, see Appendix A.2.3
- Backend telemetry including prompts, input/output token counts, selected mode, energy usage estimates, feature toggles, and navigation events

Logs were stored in Azure Cosmos DB and retained only until 31 December 2025. All infrastructure was hosted in the EU.

2.4.5 Procedure

The procedure of the whole experiment was the following:

Day 0 – Onboarding: Participants completed an informed consent form detailing data categories, retention period, and their right to withdraw. They also filled out a baseline questionnaire and received a remote walkthrough of the prototype.

Days 1–5 – Usage period: Participants used the chatbot during their daily work, received a morning reminder, and completed a four-item check-in survey in the evening.

Day 5 – Wrap-up: Logging was disabled and participants completed a comprehensive final questionnaire.

The study was approved through internal review and followed ethical guidelines for digital research involving personal usage data and behavioral tracking.

2.5 Phase V – Data Preparation and Analysis

After the experiment period, raw logs and database contents were exported as JSON and CSV. Then they were manually cleaned and combined in an Excel spreadsheet. Data sources included token counts, selected modes, timestamps, prompt content, page visits, and user preferences. To analyze the data Microsoft Excel and MATLAB were used.

The following methods were central to our data analysis:

- **Grouped Analysis:** Aggregated prompt counts, token usage, and energy consumption in different groups, per user, day and chat mode.
- **Daily Trends:** Prompt and page visit activity tracked over five days to observe usage patterns.
- **Prompt Growth Curves:** Normalized input-editing traces to assess user behavior prior to prompt submission.
- **Energy Efficiency:** Calculated Wh per 1,000 input tokens per mode (excluding fixed overhead) to compare efficiency.
- **LOWESS Regression:** Used to smooth input/output token scatter plots and analyze prediction trends.

The methodology combines survey research, prototype engineering and a five-day experiment to examine the impact of different UI-driven features regarding increasing energy awareness in conversational AI. The following chapter reviews existing work in related areas to provide context for our approach and highlight the research gaps this thesis aims to fill.

3 State of the Art

Building on the challenges identified in the problem analysis, in this section we review current research/literature and practical developments related to the three core gaps: User awareness (Perceptual Gap), technical constraints (Technical Gap), and design incentives and implementation (Design Gap). For each dimension, we synthesize insights from academic literature, existing tools, and our own survey to establish the research context and identify areas for innovation.

3.1 Perceptual Gap: User Awareness and Behavior

A growing body of Sustainable-HCI and digital energy literacy research shows that users consistently underestimate the energy consumption and environmental impact of digital services. For instance, Preist et al. [8] found that only 17% of UK participants could estimate the energy cost of uploading a photo to the cloud within an order of magnitude. Walters et al. [17] demonstrated that awareness can be improved through visible, real-time carbon-footprint feedback via the *Purple* interface, although this did not reliably lead to sustainable behavioral changes. Similarly, Chen et al. [18] observed that anthropomorphic cues in chatbots increase user engagement and affect visual attention patterns. While their study did not examine sustainability directly, it suggests that enhancing engagement through design does not automatically translate into sustainable behavior.

These findings are echoed also in broader HCI research. Geelen et al. [13] showed that real-time feedback can improve energy awareness but only has an effect when the information is easy to understand and directly actionable. HCI literature consistently finds that *just-in-time*, personalized, and contextual feedback is more effective than generic sustainability messaging [14], [19].

Our own survey confirms the presence of a significant "intention–action" gap. While most participants express concern for sustainability, only a minority are willing to set personal usage limits or pay for energy offsets. However, strong support exists for transparency and optional eco-modes, suggesting that users are open to behavioral nudges when paired with credible and interpretable feedback mechanisms.

Research outside the AI domain highlights additional intervention mechanisms:

- *Real-time feedback*: Smart-meter apps displaying live usage data increased energy awareness, though savings depended on usability [13].
- *Social comparison*: Opower's neighbour reports reduced household energy use by 2–4% [20], [21].
- *Gamification and eco-badges*: Reward systems increased pro-environmental behavior in mobile contexts [22].

These insights provide a foundation for our UI design, which aims to make energy data salient, actionable, and easy to interpret.

3.2 Survey results

To further assess current user awareness of energy consumption in conversational AI, we conducted our own survey. A total of **50 participants** completed the questionnaire. The sample consisted predominantly of digitally literate respondents: 66% identified as technically advanced (e.g., developers, researchers), and 74% reported regular or heavy use of LLM chatbots such as ChatGPT, Gemini or Claude. Most worked in IT-related roles and used LLMs primarily for research, learning, or coding purposes. The majority accessed chatbots via desktop or laptop, and 38% reported using a paid license. While not representative of the general population, this technically savvy sample reflects the early-adopter audience most likely to engage with LLM technologies and therefore most exposed to their sustainability implications.

3.2.1 General Awareness and Attitudes

Participants expressed only moderate concern about environmental impacts: The statement *"I am concerned about the environmental impact of AI usage"* averaged 3.26 on a 5-point Likert scale as seen in the table Table 3.1. However, agreement was much higher for statements related to transparency and accountability. Items such as *"LLM chatbots should disclose energy usage"* and *"AI companies don't disclose enough energy data"* each received a mean rating of 4.32, indicating strong support for disclosure even in the absence of precise knowledge.

Item	Mean	Std. Dev.
LLM chatbots should disclose energy usage	4.32	0.84
AI companies don't disclose enough energy information	4.32	1.02
I am concerned about the environmental impact of AI usage	3.26	1.19
It is important to see energy consumption information	3.20	1.18
Sustainability is important in chatbot design	2.82	1.12

Table 3.1: Awareness-related Likert Items (1 = Strongly Disagree, 5 = Strongly Agree)

These results were further supported by correlation analysis. For instance, support for an "Eco Mode" was strongly correlated with a preference for low-carbon chatbot variants ($r = 0.76$), and the perceived importance of showing energy usage data correlated with general environmental concern ($r = 0.66$). This suggests that once awareness is present, users tend to support adaptive or energy-saving features.

3.2.2 Comparative Estimation and Perception

When asked to compare the energy consumption of ChatGPT to other familiar technologies, responses varied substantially. Most participants estimated that a single ChatGPT query uses around 10 times more energy than a Google search, and that 20–100 queries would equal the energy cost of fully charging an iPhone 14. However, answers spanned several orders of magnitude from 10 to 10,000 times more energy, or from 20 to 1000 queries per charge indicating highly wrong beliefs.

This variation points not only inaccurate estimations but also fundamentally different mental models of digital energy use. Such inconsistencies are well-documented in Sustainable HCI literature: Most users lack the mental calibration to estimate the energy cost of digital services within even an order of magnitude [8]. Crucially, this uncertainty appears to be self-recognised. The strong support for transparency (Table 3.1) suggests that participants are aware of their own informational gaps even when they self-report being "aware."

This epistemic uncertainty is not entirely misplaced. The per-query energy cost of LLMs is itself an estimate, typically cited at around 0.30–0.34 Wh for GPT-4o [23], [24], while a Google search is estimated at 0.04 Wh [25]. Charging an iPhone 14 requires approximately 12.68 Wh [26], placing the true equivalence at roughly 4–40 prompts. In this light, the range of user responses reflects not just cognitive bias, but also a rational response to the lack of accessible and verifiable information. These findings highlight the value of visual and contextual feedback tools that bridge the perception gap regardless of precise technical measurements.

3.2.3 Qualitative Insights

Open-ended responses further illustrated the general uncertainty and lack of strategies. While some participants proposed fallback mechanisms (e.g., "route simple queries to Google") or limitations on

excessive use, most responses were left blank or non-informative (e.g., "–", "n/a"). This supports the interpretation that even among technically proficient users, baseline conceptual understanding of energy implications is limited.

3.2.4 Implications

The survey results confirm a significant gap between perceived and actual awareness. While users believe they understand the environmental costs of chatbot usage, their comparative estimations are widely scattered. At the same time, they strongly support transparency and feature based nudges suggesting a readiness to act if actionable information were available. This supports our design rationale: By providing contextualized, UI-driven feedback (e.g., dashboards, equivalency visualizations, or mode switching), we aim to raise user awareness without requiring expert knowledge or technical detail.

3.3 Technical Gap: Measurement and Mitigation Strategies

The energy consumption of LLMs arises from training, fine-tuning, and especially inference. While early studies focused on training costs, inference now dominates total energy use due to its frequent and large-scale deployment [3]. With nearly a billion users projected by 2025 [27], the aggregate energy burden is expected to grow rapidly.

Current benchmarks estimate that a single GPT-4o query consumes 0.30–0.34 Wh [23], [24], making it roughly eight times more energy-intensive than a Google search [25]. If chatbot usage replaced search engines at scale their energy consumption could increase by 60-70 times [1].

Mitigation strategies exist at multiple levels:

(a) **Model-level**: Sparsification, quantisation, distillation, and retrieval-augmented generation (RAG) reduce FLOPs per token [28].
(b) **System-level**: Dynamic model selection, GPU frequency scaling, and batch scheduling cut inference energy by 40–60% [11], [12], [29].
(c) **Infrastructure-level**: Renewable energy sourcing, hardware reuse, and waste-heat recycling reduce lifecycle emissions [30].

3.3.1 Recent Advances

Additional tools such as MELODI [31], LLMCO2 [32], and EnergyMeter [33] enable more accurate energy attribution, though real-time, token-level feedback for end-users remains an open challenge [9], [10]. Research also explores prompt engineering and decoding strategies that can reduce energy consumption by up to 99% in some cases without compromising output quality [34], [35] The past two years have seen a surge of research on the energy and carbon footprint of LLMs, with a focus on both measurement and mitigation. Key findings from recent studies include:

- **Life-cycle and System-level Solutions:** [3] identifies eight major life-cycle phases for LLM-powered chatbots, with hardware manufacturing and training as the most energy-intensive. System-level solutions include dynamic reporting, extended producer responsibility, and management standards.
- **Inference Optimization:** Frameworks like SPROUT [29] and DynamoLLM [12] demonstrate 40–60% carbon savings by controlling output verbosity, dynamic instance scaling, and GPU frequency adaptation. Workload-based models [36] and energy-aware routing [37] further optimize inference energy.

- **Measurement and Modeling:** Tools such as MELODI [31] and LLMCO2 [32] provide accurate, real-time energy monitoring and prediction, revealing that larger models can consume 100x more energy per token than smaller ones. EnergyMeter [33] and other benchmarks [28] highlight the dominant role of GPU and the impact of batch size, quantization, and model architecture.
- **Prompt Engineering and Decoding:** Prompt engineering [34], [38] and decoding strategies [35] can reduce inference energy by up to 99% in some configurations, with minimal or even positive effects on accuracy. Assisted Decoding and stochastic methods offer practical trade-offs between quality and energy use.
- **Code Generation and Use Cases:** Studies on LLM-generated code [39], [40] show that model size, architecture, and usage patterns (e.g., concurrency, streaming) significantly affect energy efficiency. Only a small fraction of generated code is actually used, highlighting waste.
- **Hardware and Deployment:** Using older GPUs in low-carbon regions[30] and extending hardware lifetimes can reduce embodied emissions. On-device generation is far less efficient than remote inference [41].
- **Gamification and Education:** Prompt-based games [42] and eco-badges can increase awareness and promote sustainable behavior.

These advances collectively demonstrate that both technical and behavioral interventions are needed for sustainable LLM deployment.

Despite these advances, technical constraints persist: Most accurate energy measurement tools are offline, coarse-grained, or limited to academic settings. As a result, platforms rarely offer meaningful energy feedback to users. Open-source libraries such as CarbonTracker [43] and CodeCarbon [44] estimate emissions post-hoc. However, live token-level feedback for chatbots remains an open challenge [9], [10].

3.4 Design Gap: UI Feedback, Tools, and Market Landscape

Although technical and behavioral research on sustainable LLM usage has advanced, few practical applications have reached end-users. Existing tools illustrate emerging approaches:

ScaleDown [45]: Displays per-prompt CO_2 feedback and incentivizes short, efficient prompts with a visual badge.

AI Wattch [46]: Adds a live energy gauge and dashboard to ChatGPT without backend access.

Both operate purely on the frontend and rely on static assumptions for energy estimation [47]. Their adoption and effectiveness have not yet been studied in peer-reviewed contexts.

UI interventions from other domains offer transferrable patterns: Gamification, comparative feedback, equivalency visualizations, and nudges have proven effective in domains like smart metering and mobile energy management. Prompt-based games and eco-badges [22], [42] exemplify how sustainability cues can be embedded without requiring infrastructure changes.

3.5 Conclusion and Implications

While UI-based sustainability features are well established in domains such as smart metering and mobile apps, their integration into conversational AI remains nascent. Existing research and third-party tools provide valuable design patterns and initial evidence, but several gaps persist:

- There is a lack of precise, real-time energy feedback for LLM users.
- The effective energy consumption for the major chatbots is not disclosed and there is no unanimous formula or method to calculate or predict the energy consumption of inference with such a chatbot.

- The impact of UI-only interventions on actual energy consumption and user behavior is not well understood.
- Corporate incentives and technical barriers limit the adoption of transparency features.

Our work addresses these gaps by investigating UI-only strategies to improve energy transparency and promote sustainable user behavior in conversational AI. Informed by prior research and existing tools, we introduce a user-centered approach that includes a lightweight method for estimating energy consumption suitable for UI integration. The next chapter presents the conceptual framework that underpins this solution

4 Conceptual Solution

4.1 Solution Approach

To encourage users to reduce their energy consumption, a new chatbot prototype was developed that incorporates various user interface techniques. These techniques are designed either to directly reduce the energy required for inference or to indirectly impact it by increasing users awareness of their personal energy usage.

Based on insights from our user survey, academic literature, and a market analysis, several features were defined to pursue this goal.

4.2 Baseline Chatbot

To assess the impact of energy-awareness features, it is crucial to establish a robust and widely recognized baseline. Among existing LLM-based chatbots, **ChatGPT**, developed by OpenAI, stands out as one of the most well-known and widely adopted systems worldwide at the moment of this thesis.

As detailed by Dam et al. [48], ChatGPT plays a central role in the current LLM ecosystem and is frequently cited as a benchmark for usability, response quality, and system integration. Its global prominence and maturity make it an ideal reference point for designing and evaluating alternative chatbot features.

National-level data further reinforces this decision. According to a 2024 Swiss survey by Comparis, over two-thirds of the population have already used ChatGPT or comparable tools such as Google Gemini [49]. This widespread familiarity among users ensures the relevance and realism of employing ChatGPT as the baseline system.

The baseline version of our prototype replicates the core user experience of ChatGPT, including standard features such as deleting conversations and editing conversation titles and the design.

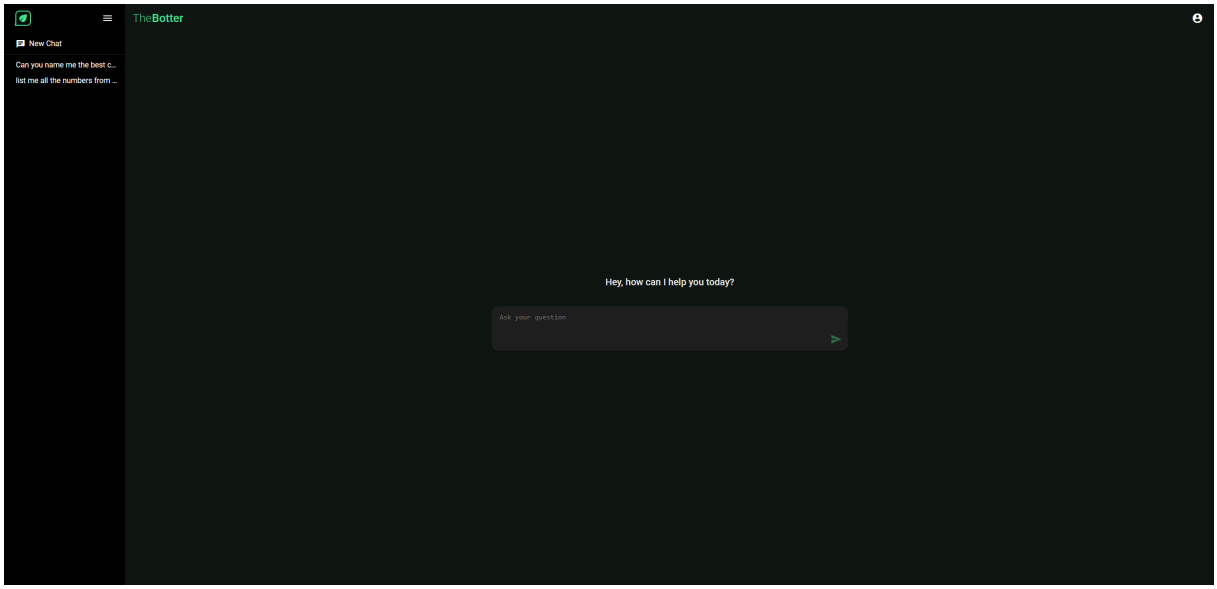


Figure 4.1: Baseline ChatGPT-inspired chatbot without sustainability features

4.3 Features and Functionalities

To promote more sustainable usage of large language models, this work explores a variety of user interface features aimed at increasing awareness of energy consumption and influencing user behavior.

The proposed features were derived through a combination of methods: A user survey conducted during the research (see Section 3.2), analysis of related academic literature, and brainstorming informed by behavioral design principles and digital sustainability research.

This section first categorizes all brainstormed features into three conceptual groups: (1) technical configuration features, (2) awareness and behavioral nudges, and (3) feedback, limits, and gamification mechanisms. Afterwards, a final subset of selected features for implementation is presented.

4.3.1 Technical Configuration Features

These features shown in the table Table 4.1 focus on reducing energy consumption through technical adjustments that are exposed to the user via the UI. They allow the system to use fewer computational resources by choosing more efficient processing strategies or limiting input complexity.

Feature	Description
Eco-Mode Toggle	Switch to a smaller model or activate multiple eco-features at once
Model Selector	Choose between performance and efficiency focused models
Ignore Context Option	Skip conversation history to reduce token processing
Alternative Tool Suggestion	Redirect queries to efficient tools like Google or DeepL
Document Splitting Warning	Ask users whether only parts of large uploads should be processed instead of defaulting to the whole document always

Table 4.1: Technical configuration features

4.3.2 Awareness and Behavioral Nudges

This category includes features as shown in Table 4.2 that aim to influence users through information, education, or subtle nudges during interaction. The goal is to increase awareness of energy usage and foster more deliberate, sustainable usage patterns in general.

Feature	Description
Prompt Energy Prediction	Estimate energy consumption before sending a message [50]
Hints	Display short energy-saving tips and nudges while chatting [51]
Prompt Suggestions	Recommend more efficient or specific prompts [51]
Footprint Awareness Tips	Relate usage to real-world analogies (e.g. phone charges) [50]
Onboarding Tutorial	Show introductory guide to raise awareness before first use

Table 4.2: Awareness and behavioral features

4.3.3 Feedback, Limits, and Gamification

These features in Table 4.3 provide real-time or retrospective feedback, integrate social comparison mechanisms, and apply gamified incentives. They are designed to reinforce sustainable behavior through motivation, reflection, and competition.

Feature	Description
Usage Meter	Track energy usage per message, session, or time period [50]
Saving Meter	Show energy saved by using efficiency features [50]
Personal Limits	Allow users to set usage caps or warning thresholds [51]
Eco-Score	Assign a sustainability score based on user behavior
User Ranking	Compare individual usage with anonymized peer data [21]
Challenges	Set daily, weekly, or monthly goals for eco-friendly use
Fee System	Introduce symbolic cost to reflect energy consumption

Table 4.3: Gamification and feedback mechanisms

4.3.4 Decision Matrix - Overview of Features

The following table Table 4.4 shows a simplified scoring overview of all features we had in mind and their scores/priority. Each feature was scored based on its expected impact (awareness and energy), research, survey support and implementation complexity (note: The scores are speculative):

Feature	Awareness	Energy	Survey	Research	Complexity	Note	Total
Eco-Mode Toggle	3	3	1	1	-2	[1]	6
Alternative Tool Suggestion	2	2	1	1	-1	[2]	5
Usage Meter	3	1	1	1	-1		5
Footprint Awareness Tips	3	1	1	1	-2		4
Prompt Suggestions	2	2	0	1	-1		4
Prompt Prediction	3	1	0	1	-2	[3]	3
Document Splitting Warning	2	2	1	0	-2	[4]	3
User Ranking	2	1	1	1	-2		3
Hints	2	1	1	1	-2		3
Model Selector	2	2	0	0	-2	[5]	2
Saving Meter	2	1	0	1	-2		2
Personal Limits	2	1	0	1	-2		2
Challenges	2	1	0	1	-2		2
Onboarding Tutorial	2	1	1	0	-2		2
Eco-Score	2	1	0	0	-2		1
Ignore Context Option	1	2	0	0	-3		0
Fee System	1	1	0	0	-3		-1

Table 4.4: Decision matrix of all evaluated UI features

Note:

1. Combines model selector; intuitive, low-effort toggle.
2. Idea seems good, although extensive analysation of the prompt might add an overhead.
3. Prediction/estimation of the actual consumption or some sort of measure is a critical feature for many others including statistical features, ranking or fee.
4. Comes with overhead effort since document prompt input must be supported by the bot.
5. Can be combined with the mode toggle; may be redundant as a separate feature.

Based on the decision matrix and a feasibility analysis, five features were selected and refined to form a coherent and complementary set. These features were chosen due to their expected high impact, their contribution to the effectiveness of other selected features and their alignment with user expectations as identified in the survey. Feasibility within the scope of this thesis was also a key consideration.

4.4 Definitive Features

In the following sections, we will describe each of the final chosen features in detail.

4.4.1 Three-Mode Switch (Eco-Mode toggle)

The Three-Mode Switch is a central UI element that allows users to consciously select between three energy profiles before sending a prompt. It builds upon the idea of an eco-mode toggle and integrates functionality from a previously proposed model selector. It also includes restrictions about the included history length. This feature enables users to balance environmental impact against output quality in an intuitive way.

Each of the three selectable modes corresponds to a different underlying model and restrictions concerning history length.

- **Energy-Efficient Mode**: Utilizes *GPT-4.1-nano* (1.1 billion parameters), limited to the last 2 messages of context. Ideal for simple or repetitive queries.
- **Balanced Mode**: Uses *GPT-4o-mini* (8 billion parameters), with the last 5 messages of context. Offers a compromise between resource usage and conversational depth.
- **Performance Mode**: Employs *GPT-4o* (175 billion parameters), including the last 10 messages of context. Suited for complex reasoning or in-depth queries.

Unlike systems that auto-switch based on prompt type, this implementation leaves the decision entirely to the user, ensuring transparency and control. The switch appears inline during prompting and can be adjusted for each individual message.

4.4.2 Metrics Dashboard

The Metrics Dashboard is a central feature designed to enhance user awareness of the environmental footprint associated with their chatbot usage. It consolidates key energy and behavior related statistics into a unified visual interface. This feature responds directly to survey feedback indicating a strong desire for transparency and supports reflection on usage patterns.

Purpose and Motivation

The dashboard serves a dual purpose: (1) to increase awareness of energy consumption through quantified feedback and (2) to support behavior change through actionable insights and motivational design. Research has shown that eco-feedback systems can lead to energy savings of 4–5% on average [19], and that real-time feedback can enhance these effects by an additional 5% [14]. To increase user engagement, the system also integrates elements of gamification such as emojis, color-coded badges and progress indicators, an approach shown to positively influence sustainable behavior [22].

Displayed Metrics

The dashboard provides live and historical insights into user behavior, with metrics structured around energy tracking, token-level statistics and usage distribution across different modes. These metrics are visualized via bar charts, line graphs, and text summaries, and are computed from both client-side and backend-collected data. Table 4.5 describes each metric and its intended user-facing function.

No.	Metric	Purpose
1	Total energy usage	Provides a high-level perspective on cumulative energy consumption over time.
2	Total number of conversations	Indicates overall usage intensity, including deleted entries.
3	Total number of prompts	Helps users understand the frequency of input, beyond just the number of conversations.
4	Total input tokens	Directly correlated with energy usage; reflects the user's prompt length.
5	Total output tokens	Complements input tokens; both are used in energy calculations.
6	Delta compared to yesterday	Displays day-over-day change in energy consumption, reinforced by gamified emoji to promote behavior change [22].
7	Daily energy usage	Visualizes trends in energy consumption over time.
8	Usage per chat mode	Helps users identify which modes contribute most to their energy footprint.
9	Prompts per chat mode	Highlights the distribution of usage across different LLM configurations.
10	Prompts per day	Tracks how frequently users engage with the system.
11	Average usage per prompt	Designed to support prompt optimization by showing the typical energy cost per message.
12	Tokens per day (input/output)	Provides token-level insight into daily usage behavior.

Table 4.5: Overview of metrics shown on the user dashboard

User Interface of the Metrics Dashboard

The dashboard is accessible via the navigation drawer and is available at any time during a session. The layout prioritizes simplicity and readability: Key metrics are displayed numerically, supported by trend indicators and visual charts. Comparative badges and emoji feedback are used sparingly to avoid overload, while still encouraging reflection and sustainable engagement.

4.4.3 Prompt Prediction

Prompt prediction is a key anticipatory feedback mechanism that estimates the energy consumption of a prompt before it is sent. This feature directly supports the broader goals of the Metrics Dashboard by enabling more reflective and energy-aware behavior at the point of decision-making. Rather than relying solely on post hoc analytics, users are provided with information that helps them evaluate the impact of their actions *before* the energy is consumed.

This aligns with findings from behavioral science which suggest that predictive or real-time eco-feedback is more effective than delayed feedback in promoting sustainable behavior [14]. By surfacing environmental costs early in the interaction, the system empowers users to rephrase or reconsider their input whether by simplifying, shortening, or selecting a more energy-efficient mode.

Functionality

While text is entered in the prompt input (but before the submission), the system constantly calculates and displays an estimated energy cost based on:

- The selected mode (Energy-Efficient, Balanced, Performance)
- The input token count (estimated via pre-tokenization)

- Expected output token length

This estimate is visualized for all chat modes in real time as the user types, which enables the users to compare the modes for their specific prompt. The calculations and the energy model coefficients used in the estimation process will be described in subsection 4.5.

Benefits and Integration

Prompt prediction not only enhances transparency but also enables the user to take proactive steps toward lower-impact usage. The feature is tightly integrated with both the Three-Mode Switch (for dynamic updates based on model selection) and the Metrics Dashboard (which records both predicted and actual energy usage for comparison). Over time, this may help users build an intuition for the cost-effectiveness of different prompt styles.

4.4.4 Energy-Note (Per-Response Footprint)

The Energy-Note feature provides immediate, post-interaction feedback about the energy consumed to generate a specific response. In contrast to the Prompt Prediction (which estimates energy usage before sending), the Energy-Note confirms and reinforces the environmental cost after the system has generated an output.

Purpose

The goal of the Energy-Note is to foster reflection through concrete, real-time feedback. After each assistant message, users are shown the exact energy cost for the current exchange, expressed both in watt-hours (Wh) and in a relatable real-world comparison further explained in 4.4.5. This reinforces the connection between digital interactions and physical resource usage.

Such fine-grained, moment-of-use feedback is designed to make energy consumption more tangible and personal, encouraging behavior change over time. The feature complements cumulative metrics in the dashboard and enhances user understanding of the impact of individual prompts and shows the impact of the history length to the energy consumption of a prompt.

User Interface of the Energy-Note

The Energy-Note appears below each assistant message in the form of a compact text element, styled subtly to maintain visual hierarchy. A tooltip or expandable section optionally provides additional context or references. This lightweight integration ensures the information is visible but not intrusive, maintaining conversational flow while offering transparency.

4.4.5 Energy Analogies

This feature originates from the concept of “Footprint Awareness Tips” and aims to make energy consumption more relatable by expressing watt-hour values in terms of everyday devices and scenarios. Pure numeric feedback in Wh may lack intuitive meaning for most users; thus, this feature provides personalized analogies to improve comprehension and emotional resonance.

It integrates with other features such as Prompt Prediction and Energy-Note to display real-time feedback using the user’s selected analogy unit. It is also embedded into the onboarding flow to encourage early reflection and engagement.

Design and Workflow

To find an analogy the user can relate to a short on-boarding process which is done by every user after the registration. During this process users are presented with several real-world energy anchors and are asked to estimate the energy cost of each. These initial guesses are used to determine which units users understand best. After the estimation users are presented with the actual consumption in Wh and their deviation in percentage, allowing them to pick the one that feels most intuitive.

Available Analogy Units

The following energy anchors are provided during onboarding and remain available for later customization:

- **iPhone 14 full charge** (12.68 Wh) [52]
- **One minute powering a refrigerator** (0.21 Wh) [53]
- **One minute working on a laptop** (0.75 Wh) [54]
- **One hour of a 500-lumen LED lamp** (6 Wh) [55]
- **One minute of PlayStation 5 gaming** (3.5 Wh) [56]

Once selected, this energy unit is used consistently throughout the whole app wherever energy feedback is shown, including next to each response (Energy-Note), in the Prompt Prediction label, and in the Metrics Dashboard summaries. This consistency ensures that users develop an intuitive mental model of their cumulative impact.

Users can change their analogy unit at any time via the settings menu, allowing them to adjust the framing as their preferences evolve.

4.5 Estimation Model of Energy Consumption

In order to calculate and predict the energy consumption of prompts we built a formula based on the current research and existing benchmarks. To quantify the energy consumption E (in watt-hours, Wh) for a single prompt–response interaction, we adopt the following linear model:

$$E = \alpha \cdot T_{\text{in}} + \beta \cdot T_{\text{out}} + \zeta$$

where T_{in} and T_{out} are the number of input and output tokens respectively. The coefficients α and β (in Wh/token) represent the energy cost per token, while ζ is a fixed overhead cost per request accounting for memory allocation, logging, and general infrastructure load.

4.5.1 Empirical Justification

Recent research confirms the viability of token-based energy estimation. Poddar et al. highlight a strong correlation between token count and inference energy [9], while Fernandez et al. criticize FLOP-based approaches and advocate for token-level modeling instead [10]. These results support the use of static coefficients α, β, and ζ.

Several empirical studies have shown that the energy cost of generating output tokens significantly exceeds that of processing input tokens. This is primarily due to the autoregressive nature of large language models, where each output token requires sequential computation and additional memory operations. Hardware-level profiling and benchmark studies consistently report that output tokens consume between 4 and 5 times more energy than input tokens [57].

OpenAI's own API pricing reflects this asymmetry: Output tokens are generally priced at 4–5× the rate of input tokens. For instance, GPT-4o charges \$5.00/M (M = millions) input tokens vs. \$20.00/M output tokens which is a 4× ratio. [58] Based on both empirical findings and pricing structure, we adopt a fixed scaling factor of:

$$\beta = 4\alpha \tag{4.1}$$

This conservative ratio balances observed energy benchmarks and commercial token economics, while simplifying model calibration.

4.5.2 Scaling Across Modes

To generalize the model across chat modes, we use OpenAI's official pricing as a proxy for computational cost. Table 4.6 shows the cost per million tokens and derived scaling ratios.

Model	Input Price (USD)	Output Price (USD)	Input Ratio	Output Ratio
GPT-4o	\$5.00 /M	\$20.00 /M	1×	1×
GPT-4o-mini	\$0.60 /M	\$2.40 /M	0.12×	0.12×
GPT-4.1-nano	\$0.10 /M	\$0.40 /M	0.02×	0.02×

Table 4.6: Model pricing per million tokens and cost-based scaling

4.5.3 Reported Energy Benchmarks

Table 4.7 lists published values for energy use per query, showing consistent differences in efficiency across models.

Source	Model	Energy (Wh)	Token Length Assumption
EpochAI (2024) [59]	GPT-4o	0.30	"Average ChatGPT query" (unspecified token count)
Jegham et al. (2025) [57]	GPT-4o	0.43	"Short prompt" (no token details)
Jegham et al. (2025) [57]	GPT-4.1-nano	$\approx$ 70x less than GPT-4o	Relative comparison for long prompt
Altman (2025) [24]	GPT-4o	0.34	No token count given

Table 4.7: Reported energy consumption per query and per token

4.5.4 Assumptions

To estimate α and β from a single interaction, we assume a typical short chat prompt with $T_{\text{in}} = 150$ input tokens and $T_{\text{out}} = 300$ output tokens. This is a typical assumption also made in prior benchmarks [57], [59]. While OpenAI's tokenizer documentation [60] provides illustrative examples, it does not specify statistically representative token lengths.

While exact distributions vary, multiple benchmark studies report typical ChatGPT interactions with input lengths ranging from 100–200 tokens and outputs often between 200–400 tokens. The chosen values reflect a balanced, realistic estimate for a brief user prompt and a moderately long model-generated response [57], [59].

The fixed overhead $\zeta = 0.02$ Wh accounts for infrastructure-related energy use that occurs independently of token processing, such as authentication, API gateway routing, logging, and request serialization. These operations introduce a baseline energy cost per query that does not scale with token count or model size.

We select this value based on empirical calibration: For example, EpochAI reports 0.30 Wh per GPT-4o query [59], making 0.02 Wh a plausible 7% share. Additionally, Jegham et al. emphasize that total query energy includes non-model components like orchestration and monitoring [57], further supporting a fixed infrastructure cost in end-to-end modeling.

This fixed overhead avoids underestimating energy use in short prompts and lightweight models while preserving a simple, transparent estimation framework. As more information about the underlying system and infrastructure is exposed this coefficients could be fine-tuned.

4.5.5 Coefficient Estimation

We derive the coefficients α, β, and ζ using two independent benchmarks for GPT-4o: The public estimate from EpochAI [59] and the infrastructure-aware analysis by Jegham et al. [57]. We assume $T_{\text{in}} = 150$, $T_{\text{out}} = 300$, and a fixed overhead $\zeta = 0.02\,\text{Wh}$. The total energy is modeled as:

$$E = \alpha \cdot T_{\text{in}} + \beta \cdot T_{\text{out}} + \zeta, \quad \text{with } \beta = 4\alpha$$

EpochAI reports $E = 0.30\,\text{Wh}$. Solving:

$$0.30 = \alpha(150 + 4 \cdot 300) + 0.02 \Rightarrow \alpha = \frac{0.28}{1350} \approx 0.000207\,\text{Wh/token}$$
$$\Rightarrow \alpha \approx 0.21\,\text{mWh/token}, \quad \beta = 4\alpha \approx 0.83\,\text{mWh/token}$$

Jegham et al. report $E = 0.43\,\text{Wh}$. Solving:

$$0.43 = \alpha(150 + 4 \cdot 300) + 0.02 \Rightarrow \alpha = \frac{0.41}{1350} \approx 0.000304\,\text{Wh/token}$$
$$\Rightarrow \alpha \approx 0.30\,\text{mWh/token}, \quad \beta = 4\alpha \approx 1.22\,\text{mWh/token}$$

Method	α (mWh/token)	β (mWh/token)	ζ (Wh)
EpochAI	0.21	0.83	0.02
Jegham	0.30	1.22	0.02

Table 4.8: Comparison of estimated coefficients for GPT-4o

We present both estimates to illustrate the range of plausible energy use under different assumptions. While Jegham et al.'s value reflects a more infrastructure-aware upper bound, we adopt the coefficients derived from EpochAI for several reasons: They align better with short, interactive usage typical in chat applications, rely on transparent and reproducible assumptions, and avoid potential overestimation due to infrastructure inefficiencies. Moreover, the resulting model remains simple and interpretable for end-user communication.

$$\boxed{\alpha = 0.21\,\text{mWh/token}, \quad \beta = 0.83\,\text{mWh/token}, \quad \zeta = 0.02\,\text{Wh}}$$

4.5.6 Final Coefficients per Mode

We scale the token-based coefficients α and β across chat modes using the pricing-derived ratios shown in Table 4.6. However, the fixed overhead ζ is assumed to remain constant across all modes, reflecting infrastructure-level costs such as authentication, logging, and request routing that do not scale proportionally with model size. This yields the final coefficients shown in Table 4.9.

Mode	**Model Name**	**Scaling**	α (Wh/token)	$\beta = 4\alpha$ (Wh/token)	ζ (Wh)
Performance	`gpt-4o`	1.00×	0.00021	0.00083	0.020
Balanced	`gpt-4o-mini`	0.12×	0.0000252	0.0001008	0.020
Energy Efficient	`gpt-4.1-nano`	0.02×	0.0000042	0.0000168	0.020

Table 4.9: Final energy model coefficients per mode

4.5.7 Predicting Output Tokens

To enable energy estimation prior to execution, the number of output tokens must be predicted from the input. While actual output length depends on prompt complexity, model behavior and context, studies show that outputs of large language models (LLMs) like ChatGPT are typically at least as long as their inputs and often even longer especially in open-ended tasks such as summarization or explanation [58], [61]. To interpret the benchmarks we assumed a input-output-ratio of 150/300 respectively 150/500, however those values were chosen under the assumption that inference does not include previous context. To account for a conversational scenario, we define the output token count as:

$$T_{\text{out}} = T_{\text{in}}$$

This identity function avoids overestimation and makes it simple. In multi-turn interactions, users frequently re-prompt or refine previous queries, causing the cumulative input (i.e., context/history) to grow. Since the frequency and extent of such re-prompting vary greatly and are hard to predict, assuming $T_{\text{out}} = T_{\text{in}}$ provides a balanced and stable estimate. Prior heuristics using 2–3$\times$ the input may overfit to specific scenarios and lack the ability to be generalized.

5 Implementation

In this chapter we present the detailed composition and implementation of the prototype. [1] [2]

5.1 System Architecture

The prototype is designed as a web application, accessible at `https://thebotter.com`, and implemented as a cloud-native system, fully hosted and scalable within the Microsoft Azure Cloud [62]. Continuous integration and deployment (CI/CD) are realized via GitHub Actions [63], ensuring seamless and automated deployments.

The frontend is a single-page application (SPA) [64] deployed via Azure Static Web Apps [65]. The backend is implemented using serverless Azure Functions [66]. Backend logic accesses Azure Cosmos DB Containers for storage and interacts with the OpenAI API [67]. User authentication is handled via Microsoft Entra ID using Easy Auth [68].

All user data is fully partitioned to ensure isolation between tenants. Administrative monitoring and logging are implemented using Azure Application Insights [69]. Manual logging is additionally implemented for experimental evaluation.

Two environments are available: A production environment for the experiment and a development environment for local testing.
The following figure gives a full overview of all the components and their interaction.

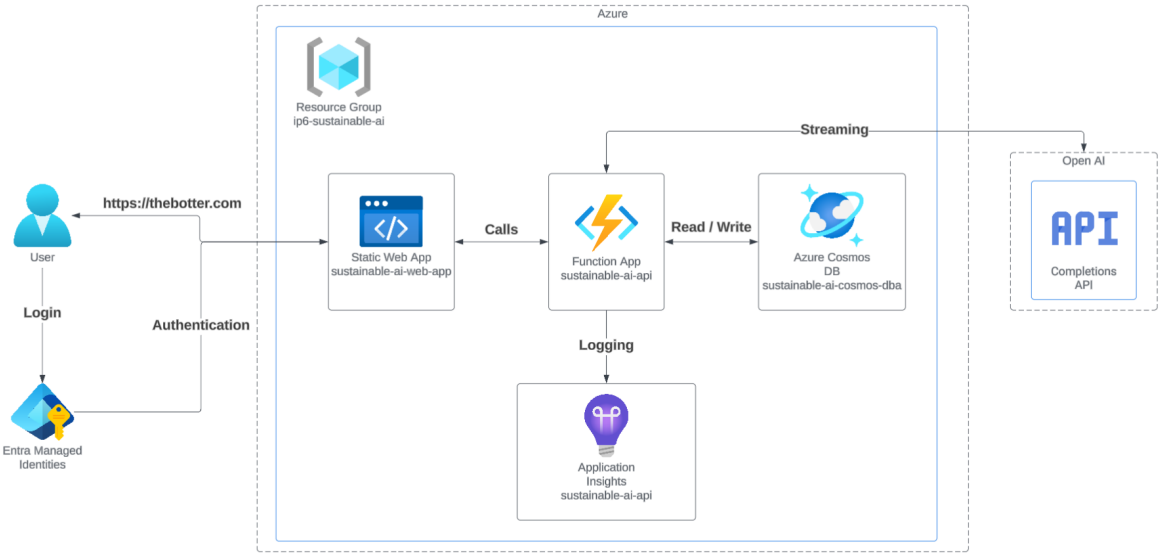


Figure 5.1: Application architecture overview

5.2 Frontend

The frontend is implemented as a single-page application (SPA) using Angular 19 [70], following the official Angular Style Guide and deployed using Azure Static Web Apps. The application is optimized for desktop use and adopts a green and black color scheme to visually reinforce the sustainability theme, with black also reducing power draw on OLED displays. [71]

[1] Frontend Repository: `github.com/simonluescherfhnw/ip6-sustainable-ai-frontend`
[2] Backend Repository: `github.com/simonluescherfhnw/ip6-sustainable-ai-backend`

All user routes are protected by Angular route guards. Authentication is handled via Azure Static Web Apps using Microsoft Entra ID (formerly Azure Active Directory). Authenticated user information is available via the `/.auth/me` endpoint, with login and logout triggered through `/.auth/login/aad` and `/.auth/logout` respectively [72].

5.2.1 Libraries and Packages

Key libraries as seen in Table 5.1 used in the frontend include Angular Material for UI components and Chart.js for data visualization:

Package	Version	Description
@angular/*	19.1.0	Core Angular packages [73]
@angular/material	19.1.0	Used as component library and for theming [74]
chartjs	4.4.9	Rendering graphs for the metrics pages [75]
marked	15.0.12	Converting markdown chat responses to HTML [76]
rxjs	7.8.0	Implements observer pattern in JavaScript, used for state management and interaction with Angular reactive forms [77]
jasmine-core	5.5.0	Jasmine test framework core library
karma	6.4.0	Test runner for Angular and Jasmine

Table 5.1: All important frontend packages

5.2.2 Code Structure

Figure 5.2 shows the folder structure of the frontend. The UI components are all directly in the app folder and split for separation of concerns.

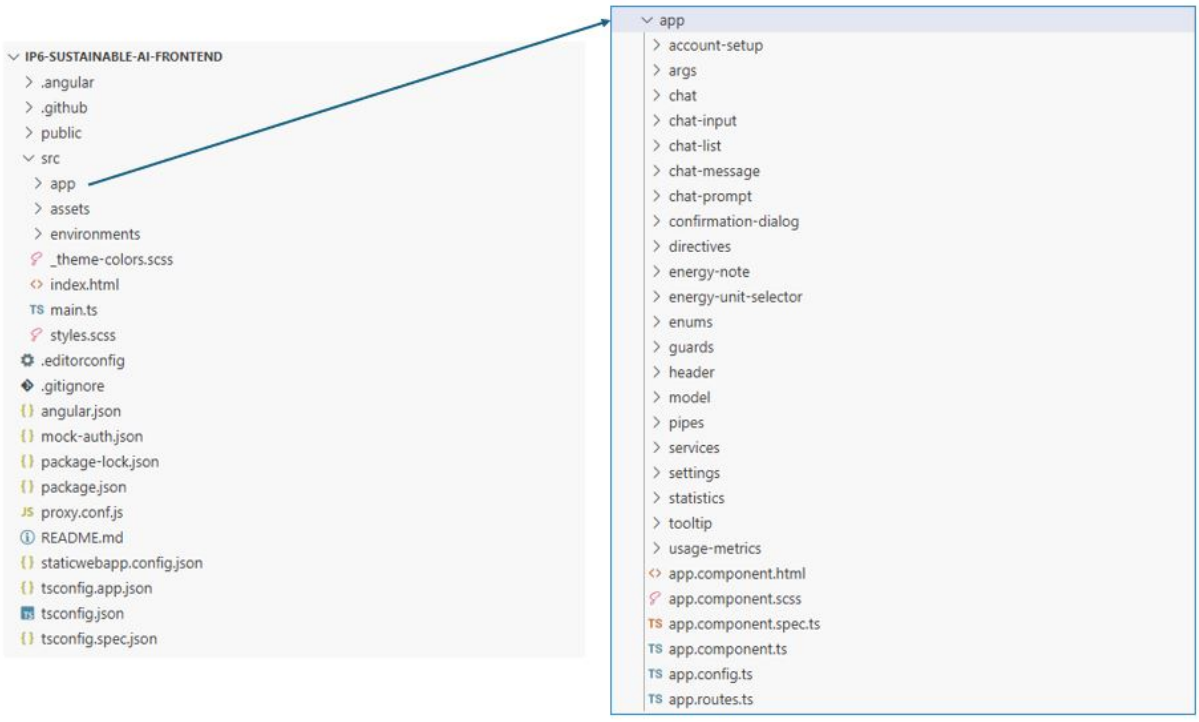


Figure 5.2: Frontend project structure

5.2.3 Routing

There are four main areas of the web application accessible via navigation. **/ or /chat** and **/chat/id** are redirecting the user to the main area, the chat section. **/metrics** shows the user's dashboard and **/settings** the settings page. After registrations users get redirected to **/account-setup** where the account initialization process starts. Figure 5.3 shows the detailed configuration of the routes in the Angular project.

```
export const routes: Routes = [
    { path: '', redirectTo: '/chat', pathMatch: 'full' },
    { path: 'chat', redirectTo: '/chat/', pathMatch: 'full' },
    { path: 'chat/:id', component: ChatComponent, canActivate: [authGuard] },
    { path: 'metrics', component: UsageMetricsComponent, canActivate: [authGuard] },
    { path: 'settings', component: SettingsComponent, canActivate: [authGuard] },
    { path: 'account-setup', component: AccountSetupComponent, canActivate: [authGuard] }
];
```

Figure 5.3: app.routes.ts

5.2.4 API Connection

On the productive environment the static web app is connected to the azure functions backend directly via configuration. It is setup so that all calls to */api* are redirected to the functions url. This also makes it possible to automatically authenticate the user for the api and attach the user claims to the request headers automatically [72]. To replicate this behavior when developing in a local environment some additional steps are necessary.

A proxy (Figure 5.4) is redirecting all requests to */api* to *http://localhost:7028/api* where the functions are running locally. Additionally, the *x-ms-client-principal* header must be set manually for every request since it wont be added automatically.

```
module.exports = {
    "/.auth/me": {
        "target": "http://localhost:4200",
        "secure": false,
        "bypass": function (req, res, proxyOptions) {
            if (req.url === "/.auth/me") {
                res.setHeader("Content-Type", "application/json");
                res.end(require("fs").readFileSync("mock-auth.json"));
                return true;
            }
        }
    },
    '/api': {
    target: 'http://localhost:7028/api', // Azure Function runtime
    changeOrigin: true,
    logLevel: 'debug',
    pathRewrite: {'^/api': ''},
    onProxyReq: (proxyReq, req, res) => {
      // Forward the auth header if it exists
      const principal = req.headers['x-ms-client-principal'];
      if (principal) {
        proxyReq.setHeader('x-ms-client-principal', principal);
      }
    }
  }
}
```

Figure 5.4: Proxy configuration file

5.3 Backend

The backend for the prototype is implemented using Azure Functions in C#. This decision was made based on Azure Functions scalability, ease of integration with other Azure services and the serverless nature of the framework which eliminates the need for infrastructure management. The backend communicates with the Angular frontend via a clean HTTP interface, is testable both locally and through the Azure Portal and integrates seamlessly with GitHub, Cosmos DB and Application Insights for logging and monitoring.

5.3.1 Functions Overview

Azure Functions offer a lightweight and scalable way to build backend logic without managing dedicated servers. In the context of a time-boxed student project, this offered several advantages:

- **No server management:** Reduces setup and DevOps effort.
- **Auto-scaling:** Adjusts to load during the experiment and reduces costs when idle.
- **Built-in GitHub integration:** Allows fast deployment with CI/CD pipelines.
- **Easy to test:** Each function can be tested in isolation via HTTP-Client or the Azure Portal.

The following HTTP-triggered functions in Table 5.2 define the main capabilities of the backend:

FunctionName	Params	Returns	Description
GetAppData	None	DtoAppData	Returns base app data including energy units and mode configurations.
GetConversations	GetConversationsArgs	ICollection <DtoConversation>	Returns user's conversations.
GetPrompts	GetPromptsArgs	ICollection <DtoPrompt>	Returns prompts in a conversation.
GetUsageStatistics	GetUsageStatisticsArgs	DtoStatistics	Returns user's statistics for the last week.
UpdateConversation	UpdateConversationArgs	DtoConversation	Updates conversation name.
UpdateUser	UpdateUserArgs	DtoUser	Updates user preferences.
DeleteConversation	DeleteConversationArgs	None	Soft-deletes a conversation.
LogPageVisit	LogPageVisitArgs	None	Logs visited page.
PredictPromptUsage	PredictPromptUsageArgs	DtoPrediction	Predicts energy usage for a prompt.
SendPrompt	SendPromptArgs	IAsyncEnumerable <DtoMessageResponsePart>	Sends prompt and returns model response stream.

Table 5.2: Overview of implemented backend Azure Functions

5.3.2 Code Structure and Modularity

5.4 Solution Structure

The solution follows a modular folder structure that aligns with best practices for organizing .NET applications as recommended by Microsoft [78], [79]. This organization supports maintainability, testability, and separation of concerns.

```
IP6-Sustainable-AI/
|-- .gitignore
|-- README.md
|-- SustainableAI.sln
|-- .github/
|   |-- workflows/
|       |-- main_sustainable-ai-api.yml
|-- .vscode/
|-- SustainableAI.Api/
|   |-- chatmode.configuration.json
|   |-- EnvironmentHelpers.cs
|   |-- host.json
|   |-- local.settings.json
|   |-- Program.cs
|   |-- SustainableAI.Api.csproj
|   |-- Common/
|   |-- Configuration/
|   |-- Data/
|   |-- Functions/
|   |-- Properties/
|   |-- Repositories/
```

```
|   |-- Service/
|-- SustainableAI.Api.Tests/
|   |-- SustainableAI.ApiTests.csproj
|   |-- TestData.cs
|   |-- Mock/
|   |-- UnitTests/
```

Key Components:

- `SustainableAI.sln`: The main solution file that aggregates all project references.
- `.github/workflows/`: Contains GitHub Actions CI/CD pipelines for automated testing and deployment.

SustainableAI.Api/ is the core Azure Functions project and contains:

- Entry and configuration files like `Program.cs`, `host.json`, and `local.settings.json`.
- `chatmode.configuration.json`: Custom configuration to define available chat modes.
- `Common/`, `Configuration/`, `Data/`, `Repositories/`, `Service/`: Represent logical layers of the application, adhering to the Clean Architecture pattern.
- `Functions/`: Contains Azure Function entry points representing external APIs.

SustainableAI.Api.Tests/ is a dedicated project for automated tests:

- `Mock/`: Provides mocked dependencies for test isolation.
- `UnitTests/`: Contains unit tests for individual services, repositories, or functions.
- `TestData.cs`: Supplies reusable test data objects.

This structure enables clear boundaries between infrastructure, business logic, and API endpoints, improving readability and testability of the solution.

5.4.1 OpenAI Integration

To replicate ChatGPT-like behavior, the backend interacts with OpenAI's chat completion API [67]. An abstraction layer is implemented via the `IChatBotService` interface to decouple model-specific logic from the core application.

- Decoupling from the vendor: The backend could be switched to e.g. Azure OpenAI or open models.
- Streaming [80] responses using server-sent events (SSE) [81], improving UX with incremental output.

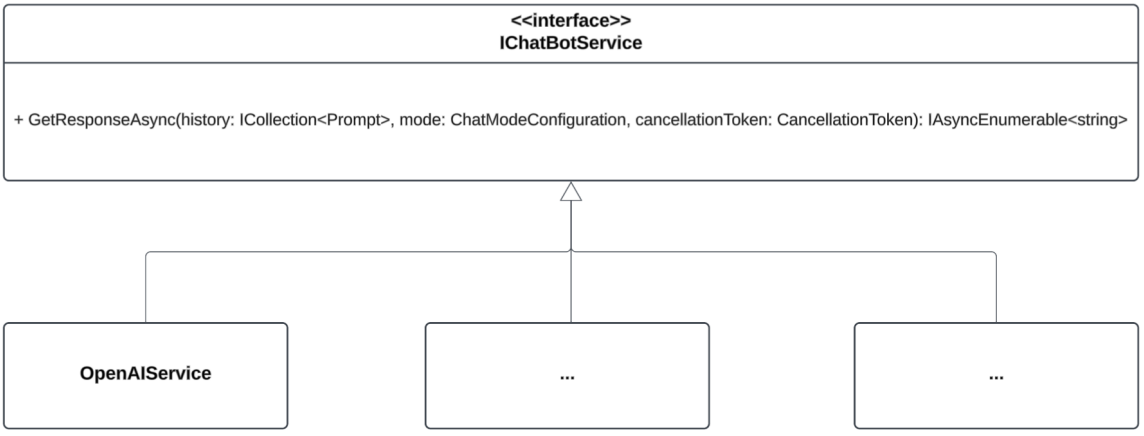


Figure 5.5: Interface abstraction for OpenAI integration

5.5 Feature Implementation

In the following sections we will explain technical implementation points of our core features.

5.5.1 Baseline Chatbot Features

The prototype replicates core ChatGPT-like functionality to serve as a reference system for measuring the effect of sustainability features. These baseline capabilities include starting new chats, renaming, and deleting existing conversations as well as prompting in existing chats or just start a new chat with a new prompt.

Users begin each session on the main chat screen (Figure 5.6) where they can initiate a new conversation. Previously saved conversations can be selected and revisited (Figure 5.7).

Each conversation includes a contextual menu that appears on hover, enabling users to rename or delete it (Figure 5.8). These changes are reflected in the Cosmos DB backend via the `UpdateConversation` and `DeleteConversation` functions.

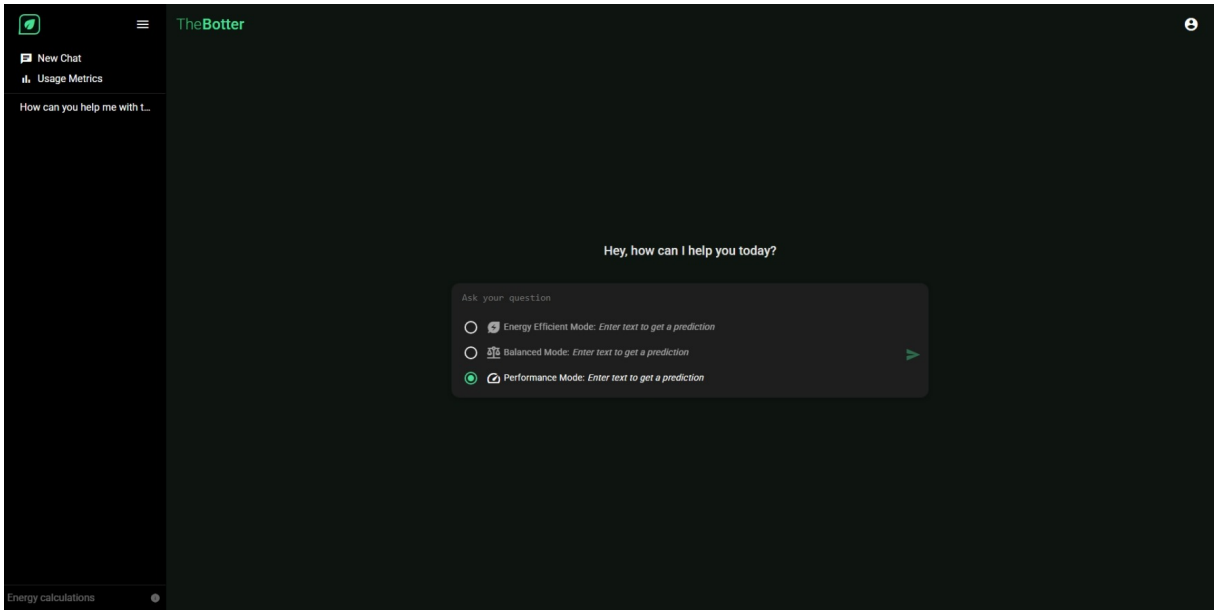


Figure 5.6: Users opening the application will land on /chat and be able to start a new dialog instantly

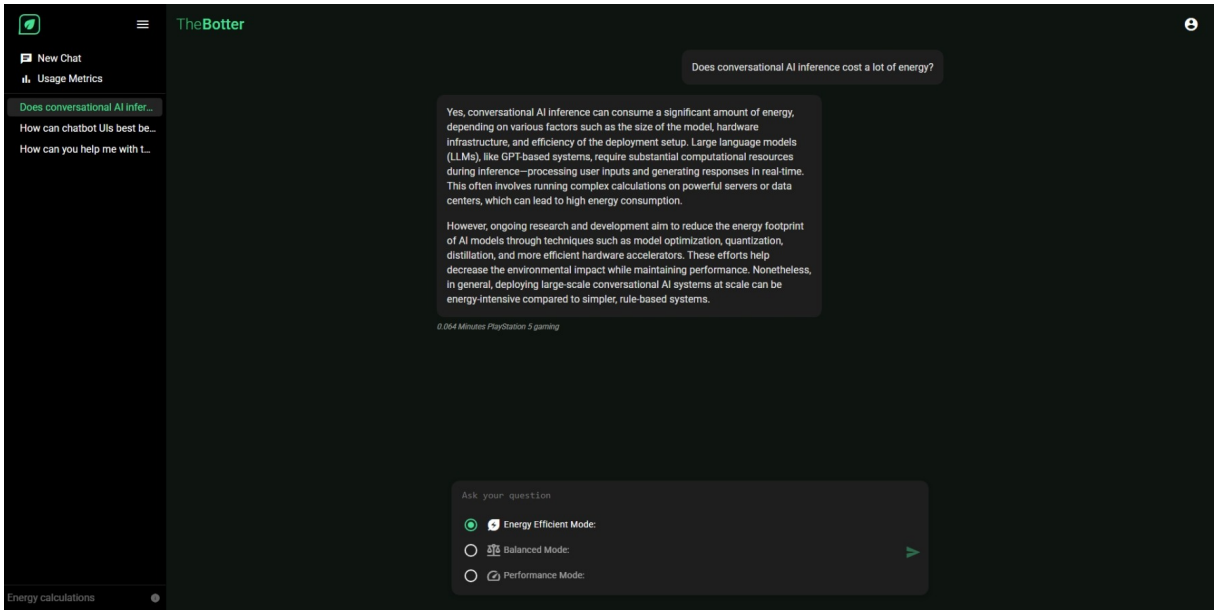


Figure 5.7: Previous conversations can be revisited to continue prompting or collect information

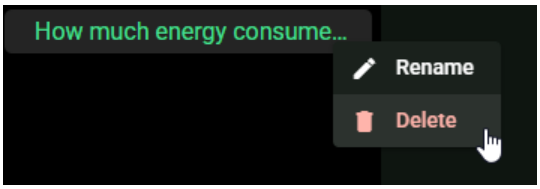


Figure 5.8: Users can delete or rename conversations via the menu icon

5.5.2 Three-Mode Switch

Implemented in the Angular chat component, the mode switch allows users to select between ”Energy-Efficient”, ”Balanced”, and ”Performance” modes. Each mode corresponds to a model preset and context window configuration. Mode changes are reflected in the frontend UI and stored per-prompt (see Figure 5.9).

The configuration is loaded from the backend via `GetAppData` and used for both prompt sending and prediction logic.

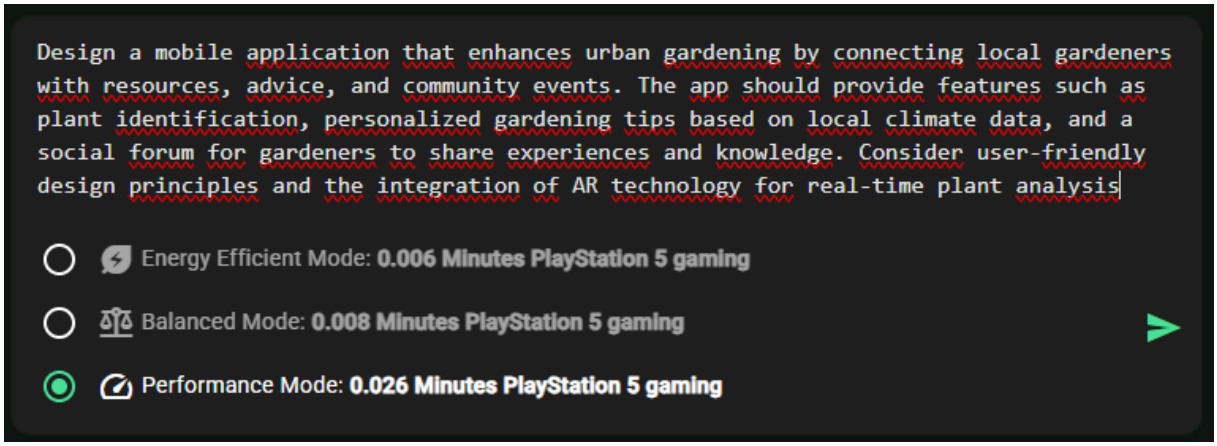


Figure 5.9: Mode can be switched for every prompt

5.5.3 Prompt Prediction

Prompt prediction is implemented via reactive data binding in the chat input component. As the user types, the frontend triggers a call to the backend function `PredictPromptUsage`, which estimates the energy consumption (in Wh) based on the input token length across all available chat modes. The resulting prediction is visualized in real time within the UI, as shown in Figure 5.10.

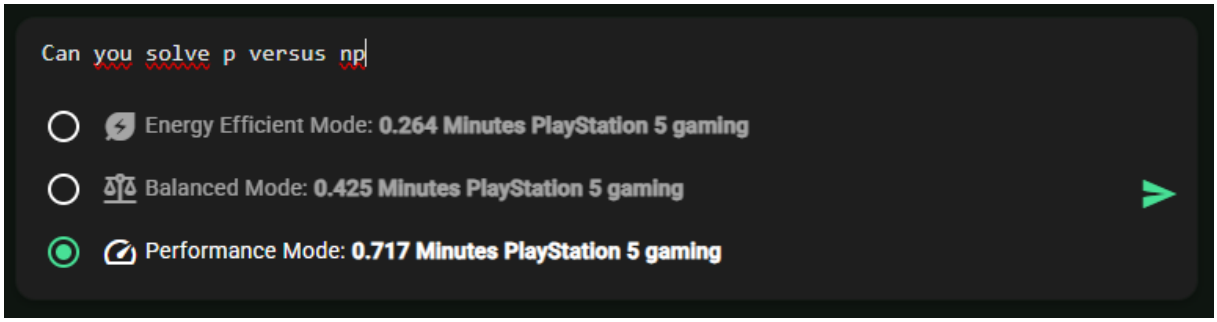


Figure 5.10: Real-time energy prediction for current prompt

The underlying estimation logic resides in the `CalculationService` class, which applies the energy estimation model introduced in Section 4.5. This service computes predicted usage by applying a linear function to the input and predicted output token counts, using coefficients provided by a corresponding `ChatModeConfiguration`. The configuration files are read from an environment variable and can be modified without redeploying the backend [82].

When a user enters a prompt, the process begins with a call to the *PredictOutputTokens* method (Figure 5.12), which estimates the number of output tokens. These values are then passed to the *CalculateUsageInWh* function (Figure 5.13), as depicted in the overall flow shown in Figure 5.11.

```
public Usage PredictUsageInWh(int inputTokens, ChatModeConfiguration cfg)
{
    if (inputTokens < 0)
    {
        throw new ArgumentOutOfRangeException("Input tokens cannot be below zero");
    }
    var outputTokens = PredictOutputTokens(inputTokens, cfg);
    return CalculateUsageInWh(inputTokens, outputTokens, cfg);
}
```

Figure 5.11: Workflow: Predicting output tokens and applying energy calculation

```
private static int PredictOutputTokens(int inputTokens, ChatModeConfiguration cfg)
{
    // Simple heuristic: 1:1
    return Math.Min(inputTokens, cfg.MaxOutputTokens ?? int.MaxValue);
}
```

Figure 5.12: Prediction of output token count based on input tokens

```
public Usage CalculateUsageInWh(int inputTokens, int outputTokens, ChatModeConfiguration cfg)
{
    if (inputTokens < 0)
    {
        throw new ArgumentOutOfRangeException("Input tokens cannot be below zero");
    }

    if (outputTokens < 0)
    {
        throw new ArgumentOutOfRangeException("Output tokens cannot be below zero");
    }

    var energyWh = cfg.AlphaWhPerInputToken * inputTokens +
                   cfg.BetaWhPerOutputToken * outputTokens +
                   cfg.ZetaConstWh;

    return new Usage
    {
        NumberOfInputTokens = inputTokens,
        NumberOfOutputTokens = outputTokens,
        UsageInWh = energyWh
    };
}
```

Figure 5.13: Energy usage calculation based on token counts

5.5.4 Energy Note

After the chatbot's response is received, the actual energy usage is calculated using the function `CalculateUsageInWh` and displayed beneath the response message. In addition to the numeric value, the UI also presents a relatable real-world analogy based on the user's selected energy unit (e.g., minutes of LED light usage), as illustrated in Figure 5.14.

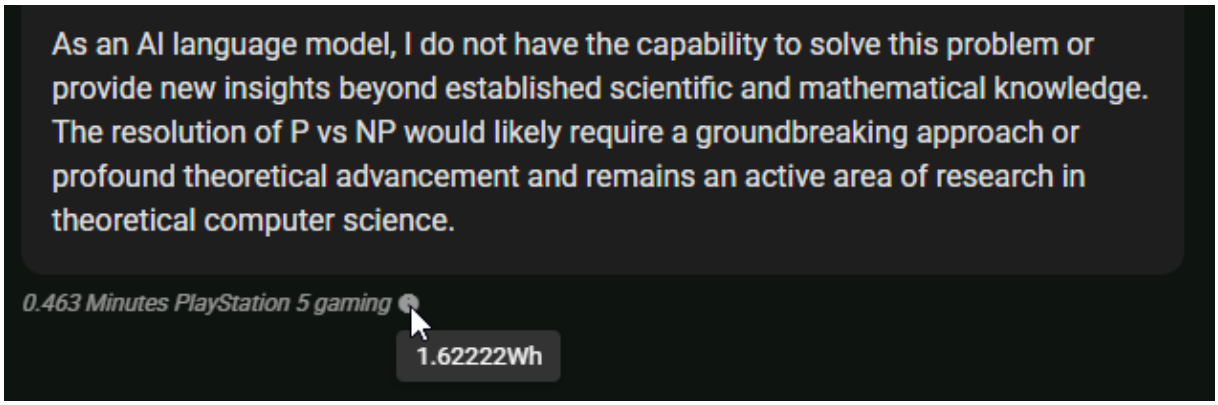


Figure 5.14: Post-response energy note with real-world analogy

5.5.5 Metrics Dashboard

The dashboard visualizes aggregated usage metrics using Chart.js. The statistics are retrieved via `GetUsageStatistics` and rendered into graphs showing energy usage, token counts, prompt counts, and usage distribution.

The frontend uses a `StatisticsService` to manage API interaction and local state. Charts include bar, line, and pie diagrams with dynamic annotations. More details on the specific metrics can be found in 4.4.2.

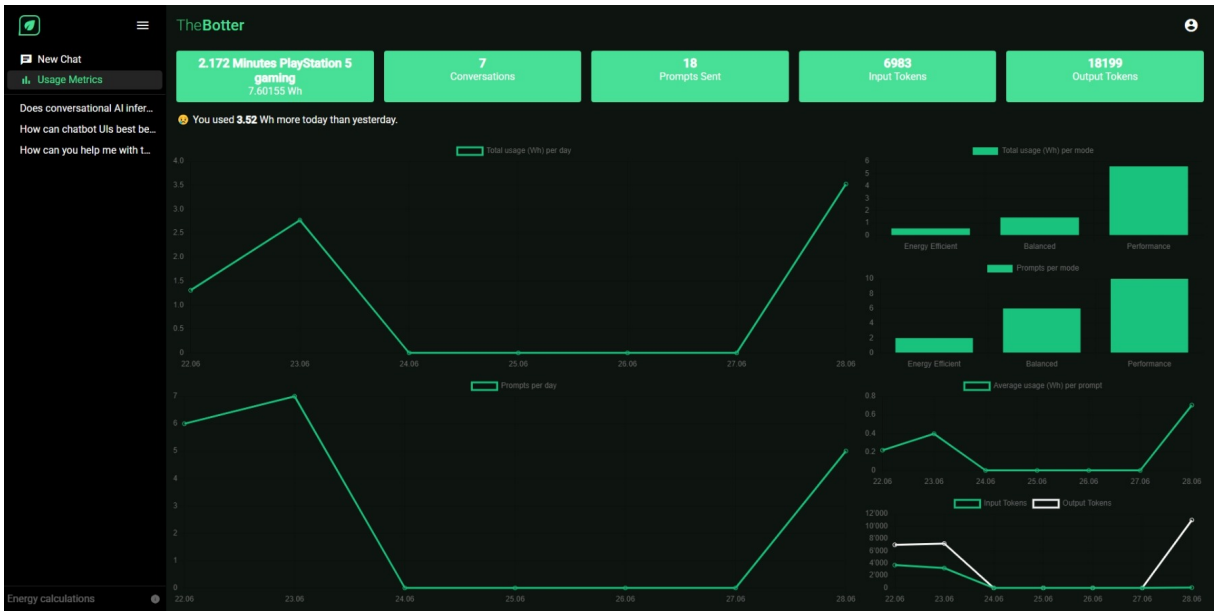


Figure 5.15: User metrics dashboard with energy, token, and mode usage breakdown

5.5.6 Energy Analogies and Onboarding

During account setup, users are shown a guessing game where they estimate the energy of real-world devices. Their responses are scored based on the deviation and used to recommend an energy anchor (e.g., LED lamp, iPhone charge). Screenshots shown in Figure 5.16 and 5.17.

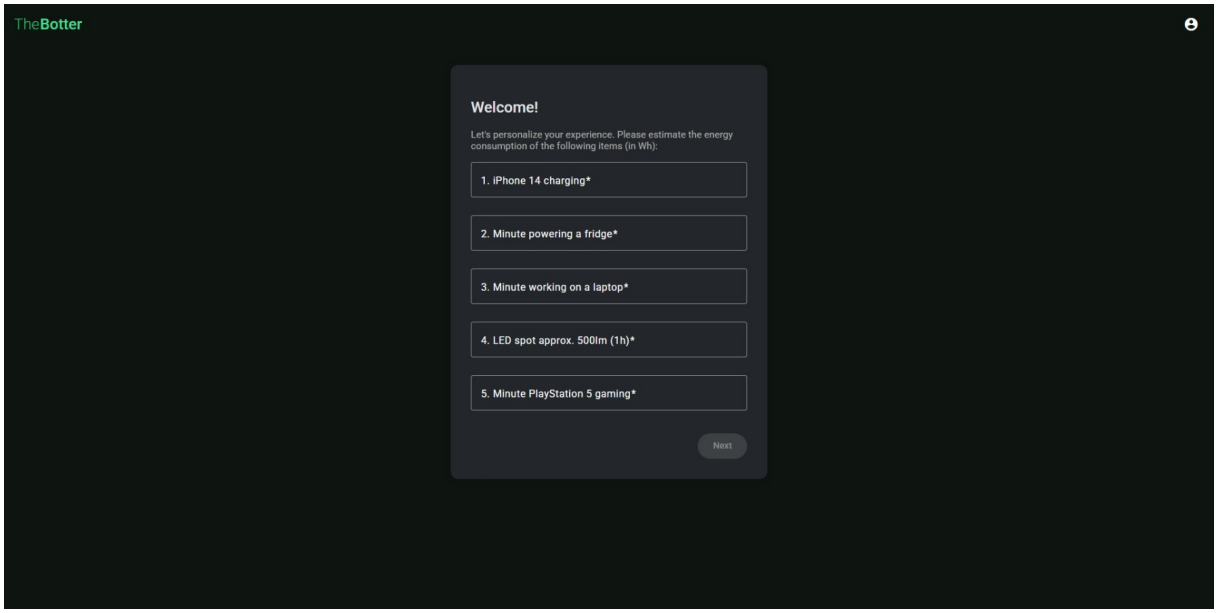


Figure 5.16: Guessing energy analogies

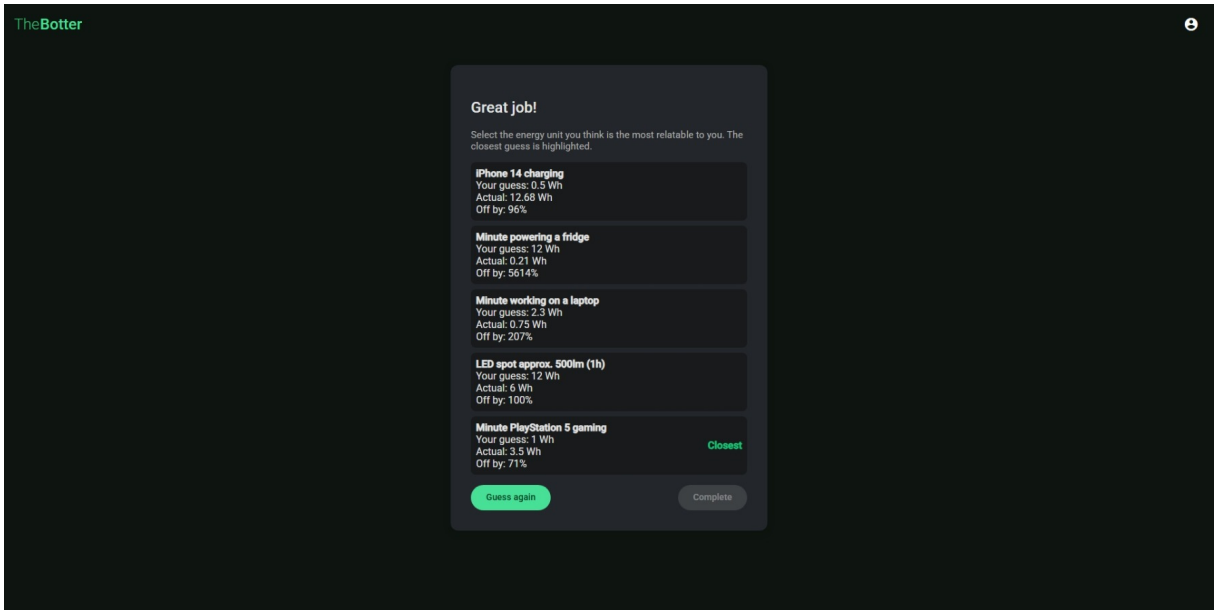


Figure 5.17: Users can choose their preferred energy analogy

This selected analogy unit is stored via `UpdateUser` and used throughout the app in predictions, notes, and the dashboard.

5.5.7 Settings Page

The settings page allows users to change their energy unit or disable features. Toggles trigger backend updates via `UpdateUser` and are also logged via log type `SustainabilityModeChange`.

Figure 5.18 shows the final implementation.

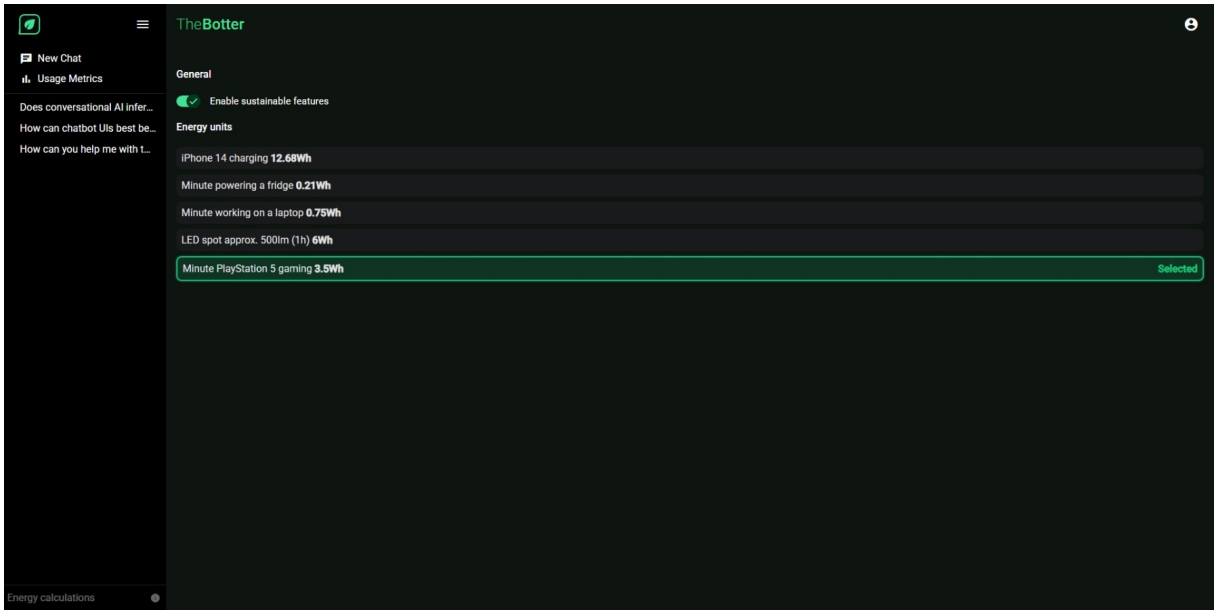


Figure 5.18: User settings

5.6 Data Storage: Cosmos DB

Azure Cosmos DB was chosen as the primary data storage solution due to its flexible, schema-less design, low-latency access, and scalability. This makes it particularly well-suited for storing chat-related data such as user prompts, responses, and metadata, especially in scenarios where the structure may evolve over time, such as future integration of Retrieval Augmented Generation (RAG) features.

As a NoSQL document database, Cosmos DB allows for heterogeneous document structures within the same container. This flexibility enables efficient storage of evolving prompt data, user metadata, and interaction histories. For example, arrays such as previous messages or model-specific metadata can be stored directly as `string[]` without requiring complex schema changes.

The application organizes its data into multiple containers. Each container is logically associated with an entity type, and partitioning is primarily based on the user ID to isolate user data and support horizontal scalability. The database schema is visualized in the following diagram. Note that foreign key relationships shown in the diagram are for conceptual clarity only, Cosmos DB does not support referential integrity across containers [83].

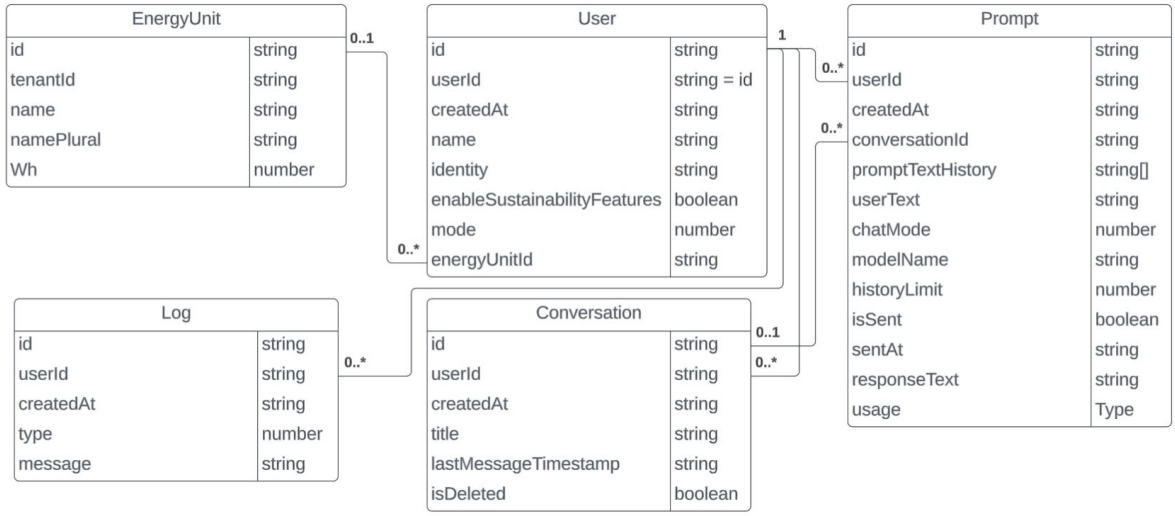


Figure 5.19: Logical database schema (conceptual relationships)

All containers, except the one for energy units, are partitioned by user ID. This design allows for scalable performance, simplified query filtering by user and logical data isolation. Although this does not enforce access control at the database level, it enables efficient per-user access filtering at the application level. The energy units container, by contrast, is globally shared across users and uses a constant partition key to avoid redundant definitions.

5.6.1 User

Each document in this container represents a user entity. The `id` field corresponds to the authenticated user ID from Microsoft login. This `id` is also used as the partition key, meaning each user has their own partition. Additionally, this user ID is referenced as the partition key in other containers.

5.6.2 Conversation

Represents a user conversation. Each conversation belongs to one user. Deletion is handled using a soft-delete strategy by marking records as deleted, rather than removing them from the database.

5.6.3 Prompt

Stores individual prompt–response pairs. A draft prompt document is created immediately when the user starts typing. If a prediction is generated, the record is persisted in the database, even if the user later cancels or changes their input. Once a prompt is sent, it is associated with a conversation, marked as ”sent”, and enriched with additional metadata such as the model response, estimated energy usage, chat mode, model name, and the included history size. This allows accurate reconstruction of the context for analysis or replay, even if configuration settings change over time.

5.6.4 Log

Used primarily for tracking user behavior during experiments. Each log entry is written manually and assigned a type. Current log types include: `Unknown`, `PageVisit`, and `SustainabilityModeChange`.

5.6.5 EnergyUnit

Represents an energy unit or analogy such as an ”iPhone 14 charge”. Since Cosmos DB requires a partition key on all containers, this container uses a fixed partition key called `tenantId` with the constant value `"0"`. This pseudo-partitioning avoids data duplication while remaining compliant with platform requirements.

5.7 Non functional requirements

Beyond implementing the core features of the web application, various non-functional aspects have been considered to ensure the prototype is maintainable, scalable and secure. This chapter discusses the system's architecture and operational qualities, focusing on deployment processes, authentication, performance capabilities and technical limitations.

While the application is not yet in a production environment, many architectural decisions were made with future extensibility in mind. The use of Azure-native services such as Static Web Apps, Functions, and Cosmos DB allows for scalable and secure operation with minimal configuration. Furthermore, modern development workflows such as continuous integration and automated deployment contribute to a high level of maintainability.

The following sections provide a detailed overview of the system's non-functional characteristics.

5.7.1 Logging and monitoring

The prototype incorporates basic logging and monitoring functionality using Azure-native tools. Azure Application Insights is connected to the Azure Functions application and provides real-time observability into backend operations. It collects telemetry such as request rates, failure rates, response times, and traces, which can be explored through dashboards, Kusto queries, or alerts.

On the database level, monitoring is available via the Azure Portal within the Cosmos DB account. Built-in tools allow developers to inspect metrics such as throughput consumption (RU/s), latency, and storage usage. In addition, logs and query performance data can be viewed to identify performance bottlenecks or data inconsistencies.

These monitoring capabilities are crucial for debugging, performance tuning, and maintaining operational health, especially in scalable, distributed environments like those offered by Azure.

5.7.2 Security

Security within the developed prototype is centered around authentication and access control, leveraging Azure Static Web Apps' built-in identity management. By default, Azure Static Web Apps supports authentication providers such as Microsoft, GitHub, Google, and Twitter. This project uses Microsoft Entra ID (formerly Azure Active Directory) as the sole identity provider [84].

When users access the application, they are redirected to a secure Microsoft login page. Upon successful authentication, Azure injects identity information into the request via the `x-ms-client-principal` header. This header is base64-encoded and contains user attributes such as ID, email, and assigned roles.

On the frontend, route-level protection is implemented using an Angular `AuthGuard`. For any navigation to sensitive routes such as chat, settings, metrics, or account setup, the guard first checks if the current user is authenticated. If the user is not authenticated, the guard redirects them to the login endpoint (`/.auth/login/aad`). This ensures that only authenticated users can access protected areas of the application.

In development mode, where authentication is not enforced by Azure, a proxy configuration is used to simulate authentication. The frontend serves mock identity data via the `/.auth/me` endpoint using a local file (`mock-auth.json`). This approach enables developers to test authenticated behavior locally without logging in via Microsoft Entra ID.

The backend, implemented using Azure Functions, independently verifies the presence and validity of the `x-ms-client-principal` header. A custom `AuthorizationService` extracts and deserializes this header, ensuring that only requests from authenticated users are processed. Unauthorized

requests, either due to a missing header, invalid deserialization, or lack of the "authenticated" role are rejected early with detailed error logging. This additional server-side validation ensures that even if a route were misconfigured on the frontend or Azure Static Web Apps, access to backend functions would remain protected.

Access control is further enforced declaratively in the `staticwebapp.config.json` configuration file, which restricts the `/api/*` route to authenticated users only. This creates a defense-in-depth model: Configuration-level protection via Azure, route guarding on the frontend, and validation on the backend.

Currently, the system adopts a flat authorization model: All authenticated users are granted identical access privileges. No role-based or fine-grained access control mechanisms have been implemented at this stage. If the application evolves toward a multi-role system (e.g., users vs. admins), Azure Static Web Apps supports assigning roles [85], and additional logic could be introduced on both frontend and backend to differentiate access.

Overall, the prototype achieves a strong level of security appropriate for its scope by combining platform-native authentication with layered access control and explicit validation. Improvements for future production deployment may include token expiration handling, detailed audit logging, support for additional identity providers, and fine-grained role enforcement.

5.7.3 Testing

Automated tests have been implemented for both the backend and frontend to ensure functional correctness and maintainability of the application over time.

Backend: The backend tests are located in a dedicated test project within the backend's repository and cover both unit and integration levels. The primary focus is on the business logic of the application. Unit tests isolate individual components and validate their behavior under different conditions. Integration tests verify that multiple components such as services and controllers work together as expected, without relying on real database connections or external services. In both cases, data access and external dependencies are mocked to ensure test speed, reliability, and independence from the deployment environment [86], [87].

Frontend: The frontend project includes a comprehensive suite of automated tests for all components containing extensive TypeScript logic and for all the services [88]. These tests are implemented using Jasmine [89] and executed with Karma [90]. The test suite is integrated into the CI/CD pipeline, ensuring that all frontend logic is verified before deployment. This approach helps maintain high code quality as the application evolves.

This test structure allows both backend and frontend logic to be verified in isolation and supports a robust, maintainable development process. Automated tests are also integrated into the continuous integration pipeline, which ensures that code changes are verified before deployment.

5.7.4 Scalability and Availability

Although scalability and availability were not primary goals of this prototype, the chosen architecture inherently provides a foundation that could support these qualities in a production environment.

The frontend is hosted using Azure Static Web Apps, which is globally distributed by default. While the deployment is configured with the location *West Europe* for backend integrations, the static content itself is served via Azure's global edge network, ensuring low-latency access from most geographic regions [91]. Currently, no advanced configuration such as custom CDN integration or regional routing has been applied. However, the platform supports enhancements such as Azure Front Door [92] and

enterprise-grade edge caching that could significantly improve performance and availability in high-traffic scenarios.

The backend logic and data storage are deployed in the Azure region *Switzerland North*. This includes the Azure Functions used to handle API requests and the Azure Cosmos DB instance managing per-user data. Azure Functions is currently running on the default consumption plan, allowing automatic scaling based on traffic demand [93]. This suits the low and intermittent workload of the prototype, although cold-start latency may become an issue in production. Premium plans are available to mitigate this by enabling pre-warmed instances and improved scaling limits [94].

Azure Cosmos DB is provisioned in single-region mode in Switzerland North. While this simplifies the setup, it also introduces a regional single point of failure. Cosmos DB supports multi-region replication with automatic failover and global distribution, which could be enabled to increase resilience and reduce read latency across geographies [95]. Additionally, Cosmos DB's autoscale and partitioning features remain unused but offer a clear path to horizontal scalability as the dataset or user base grows [96].

The application relies on the OpenAI Completions API [97], which is hosted outside of Azure and geographically decoupled from the application's infrastructure. This introduces some dependency on public internet connectivity and may be affected by regional latency or rate limits. The prototype operates well within the default OpenAI API quotas. However, for production scenarios, implementing retry logic, monitoring API usage, and applying for higher quotas would be essential [98].

In summary, while the current deployment operates under default configurations suitable for prototyping, the system is cloud-native and inherently scalable. All critical services such as Azure Functions, Cosmos DB and Static Web Apps support scaling mechanisms, multi-region deployment, and high-availability configurations that could be activated with minimal architectural changes. These capabilities would become crucial if the prototype evolves into a production-grade system [99]. Enhancing scalability and availability would primarily involve enabling geo-replication, autoscaling configurations, and CDN integration, all of which are natively supported within the Azure ecosystem [99].

5.7.5 Continuous Integration and Deployment (CI/CD)

The project incorporates two separate CI/CD pipelines. One for the frontend and one for the backend using GitHub Actions in combination with Azure services. These pipelines enable continuous integration and deployment with minimal manual intervention, ensuring that the latest versions of both the user interface and backend logic are reliably deployed to Azure upon each commit to the main branch.

The **frontend pipeline** is defined in the file `azure-static-web-apps-green-mud-04afae203.yml`. It is automatically triggered when changes are pushed to the frontend directory. The pipeline installs dependencies, builds the Angular application, runs the tests and deploys it to Azure Static Web Apps. This deployment process is fully integrated into GitHub via the Azure Static Web Apps GitHub Action [100], which abstracts away much of the manual configuration typically required.

The **backend pipeline**, defined in `main_sustainable-ai-api.yml`, is responsible for building the .NET project, running all automated tests, and deploying the application to Azure Functions. This pipeline ensures that only tested code reaches the production environment. It uses standard GitHub Actions for .NET development, such as building, testing, and publishing the application, along with the Azure Functions GitHub Action for deployment [101].

While the CI/CD setup ensures automated deployments, the project currently does not include any form of **Infrastructure as Code (IaC)**. All Azure resources have been provisioned manually via the Azure Portal.

This decision reflects the prototypical nature of the project. Since the goal was rapid iteration and experimentation, setting up full IaC automation would have added complexity and overhead without immedi-

ate benefit. Manual provisioning allowed for flexible configuration and faster development during early stages which was beneficial in a rapidly evolving prototype setting. However, for production scenarios, IaC is considered a best practice to enable reproducible, version-controlled, and automated infrastructure management [102], [103].

Overall, the CI/CD approach implemented in this project provides a pragmatic balance between automation and flexibility. It ensures reliable deployments while keeping the setup lightweight and maintainable for a prototype.

5.8 Limitations and constraints

The current prototype of the web application is subject to several limitations and constraints, which result both from deliberate design decisions and from the technical scope of the implementation.

At this stage, the web application is optimized for desktop use and has not been adapted for mobile or tablet devices. Furthermore, the user interface and chatbot interactions are available exclusively in English, which may limit accessibility for non-English-speaking users.

Interaction with the chatbot is currently limited to text input and output. Features such as voice input, file upload, or multimodal interfaces are not supported. This constraint aligns with the prototype's goal of keeping the interaction model simple while focusing on core functionality.

User authentication is restricted to login via Entry ID, with no support for alternative identity providers or anonymous access. While sufficient for testing and evaluation, this would need to be extended in a production environment to support broader identity management options.

The web application does not implement pagination or lazy loading for chat history. As a result, only the most recent 100 conversations and the last 100 messages (prompts and responses) are displayed to the user. This simplification was chosen to reduce implementation complexity but may limit usability for users with extensive chat histories.

The chatbot's responses are streamed in real-time to the frontend using server-sent events. This approach improves user experience by reducing perceived latency but limits compatibility to a subset of models that support streaming. While it would be possible to support non-streaming models by implementing a custom version of the `IChatBotService` interface that yields responses as an `IAsyncEnumerable`, such an adaptation is not yet implemented.

On the data persistence layer, the use of Azure Cosmos DB introduces another important constraint: As a NoSQL database with a flexible schema, Cosmos DB does not enforce structural consistency across documents. To mitigate this, the application ensures schema validity at the serialization level by marking key properties as required. This ensures that documents with missing or malformed data trigger exceptions during deserialization, preventing faulty runtime behavior. However, this approach only identifies the problems but it does not resolve them. A more robust solution would be to implement lazy migrations adjusting documents on-the-fly as they are loaded to conform to the latest schema version. Such an approach would improve resilience and forward compatibility and could be considered in the future if the prototype evolves into a production-grade application.

6 Validation and Results

This chapter evaluates to what extent the UI-only interventions proposed in Section 4.3 achieve their intended goals of (1) raising user awareness regarding the energy footprint of LLM queries and (2) nudging users towards lower-impact behavior. We triangulate three independent data sources:

(a) Five *daily awareness check-ins* ($n = 11$, 52 responses, 95 % completion),
(b) A *post-study questionnaire* ($n = 11$, 100 % completion), and
(c) Detailed *user data and behavioral logs* captured by the prototype backend.

6.1 Results from Daily Check-in and Final Questionnaire (a & b)

In this section we report the survey-based outcomes from the five-day experiment.[3]
We focus on two self-report surveys:

- The **daily awareness check-in** (five small surveys, $n = 11$; 52 responses; 95 % completion rate)
- The **final comprehensive questionnaire** ($n = 11$; 100 % completion rate).

All questions used a 5-point Likert scale [15] with 1 meaning *strongly disagree* and 5 *strongly agree*.

6.1.1 Daily Awareness Trajectory

Table 6.1 shows the mean self-reported awareness[4] for each study day. The small mid-week dip (Days 3–4) matches qualitative feedback that participants "stopped paying attention once the novelty wore off," suggesting repetition alone is insufficient, more salient reminders were needed until the dashboard visit on Day 5 (see below). This upward trend from Day 1 ($M = 3.27$) to Day 5 ($M = 4.44$) supports **Hypothesis H1**, which states that visibility of energy information increases user awareness over time.

Day	Mean	Std. Dev.	Δ vs. Day 1
1	3.27	0.90	—
2	3.64	0.81	+0.37
3	3.09	1.14	−0.18
4	3.10	0.99	−0.17
5	**4.44**	0.73	**+1.17**

Table 6.1: Mean awareness per day *"At this moment I'm aware of the energy cost of the prompts I sent today"*, (1 = Strongly Disagree, 5 = Strongly Agree)

6.1.2 Feature Salience

Across all 52 check-ins, participants rated the *Energy-Note* and the *Mode-Toggle* as the most visible interventions ($M = 3.19$ for both), whereas active dashboard use occurred on 59.6 % of participant-days. The spike in awareness on Day 5 coincides with the highest dashboard traffic (Table 6.5), indicating that summary feedback still plays a crucial consolidation role even after inline cues are present. This further corroborates **Hypothesis H1**, since increased visibility of features correlates with the highest awareness score.

[3] Behavioral log data are analyzed separately in Section 6.3.
[4] "At this moment I'm aware of the energy cost of the prompts I sent today"; 5-point Likert.

6.2 Final Questionnaire

6.2.1 Self-Reported Behavioral Change

Table 6.2 summarizes the behavioral section. Nine of eleven participants ($81.8\,\%$) reported *regularly* switching to Energy-Efficient-Mode when answer accuracy was "not mission-critical" ($M = 4.18$). Prompt shortening received a neutral rating ($M = 3.00$), and the detailed examination of the log data in Section 6.3 confirms there was no substantial reduction in input length on average (see Figure 6.2). Distraction by sustainability cues was low ($M = 1.45$), suggesting the interventions did not impair chatbot usability.

Item	Mean	Std. Dev.
Energy-Efficient-mode when accuracy uncritical	**4.18**	0.75
Shortened / reduced prompts	3.00	1.34
Dashboard visited $\geq 2\times$	0.77 (Yes)	n.a.
Energy info distracted me	**1.45**	0.82
Want similar features elsewhere	**3.91**	1.14

Table 6.2: Behavioral items, (1 = Strongly Disagree, 5 = Strongly Agree)

Preliminary day-level correlations show that higher awareness is associated with a greater Energy-Efficient-mode share and shorter prompts, which supports **Hypothesis H2**, stating that awareness is positively correlated with pro-sustainability behavior.

6.2.2 Per-Feature Usability

All four implemented features scored above 4 ("easy to understand and effect was clear"), with the *inline three-mode toggle* leading ($M = 4.64$). Qualitative answers (Section 6.2.3) emphasized participants' desire for more ephemeral reminders and aggregate figures in familiar units.

Feature	Mean
3-mode toggle	**4.46**
Placement of toggle	**4.64**
Energy note (per response)	4.09
Metrics dashboard	4.00

Table 6.3: Usability ratings: *"Easy to understand and effect clear"*, (1 = Strongly Disagree, 5 = Strongly Agree)

6.2.3 Qualitative Feedback – Text Answers

Four of the eleven participants wrote substantive comments, each referring to different ideas or improvements. Three themes emerged:

- **T1 Proactive reminders:** Two participants requested "occasional push notifications" to surface energy consumption without requiring a dashboard visit.
- **T2 Real-time aggregate figures in familiar units:** Three comments called for a constantly visible "current usage" badge expressed in the chosen energy unit (e.g., *Laptop-minutes*) instead of raw Wh values, which is currently only visible on the dashboard.

T3 Contextual framing of impact: Respondents stated that numbers alone "don't really help me that much" and proposed a concise "wake-up call" every 100 prompts, optionally linking to a deeper explanation or video.

One participant (original German comment) also suggested: *"Bei sehr kurzen Prompts das Verwenden einer klassischen Google Suche anstelle von LLM vorschlagen"* ("For very short prompts, recommend a classic Google search rather than an LLM"), which was also a potential feature considered but not implemented.

6.2.4 Summary of Awareness Trajectory

- Awareness increased over the week, reaching $M = 4.44$ on Day 5
- Self-reported Energy-Efficient-mode adoption was high and distraction minimal.
- All usability means exceeded $4/5$, with the mode toggle scoring highest.
- Preliminary day-level correlations show that higher awareness is associated with a greater Energy-Efficient-mode share and shorter prompts, supporting **Hypothesis H2**.

These findings motivate a deeper behavioral analysis based on the collected server log data presented next.

6.3 Behavioral Results from Application Data

6.3.1 Data Set Results Total

A total of 278 prompt–response pairs were recorded over the study's duration, totaling 683,580 processed tokens (231,779 input, 451,801 output) and 168.0 Wh estimated energy. In the established energy analogies this would correspond to:

- 13.2 full charges of an iPhone 14
- 13.3 hours of refrigerator usage
- 3.7 hours of laptop use
- 28 hours of a 500-lumen LED lamp
- 48 minutes of PlayStation 5 gaming

6.3.2 Energy-Unit Selection

Eight of eleven participants selected the *iPhone 14 charge* analogy, two preferred *minute powering a fridge*, and only one chose *minute PlayStation 5 gaming*, no one selected the laptop or LED-lamp options (Table 6.4). Participants thus gravitate toward anchors that (i) represent a *complete, familiar action* (charging a phone) and (ii) occur in their everyday routine. Time-based slices of low-power devices were perceived as too abstract. Future designs should therefore favor whole-device cycles e.g. "electric-toothbrush charge" over "x-minutes of use" to maximize intuitiveness.

Mode	Used by
iPhone 14 charging	8 Users
Minute working on a laptop	0 Users
Minute powering a fridge	2 Users
LED spot approx. 500lm (1h)	0 Users
Minute PlayStation 5 gaming	1 User

Table 6.4: Energy unit preferences

6.3.3 Dashboard Engagement

Across the five-day study, the dashboard was opened **41** times ($\bar{x} = 3.7$ per participant, median = 3), see Table 6.5 for the overview. Two participants never visited it, while the top user (ID 8) alone generated **29%** of all visits; the three most active users (IDs 8, 2, 10) accounted for **59%** (24 / 41). Day-level activity peaked on Day 1 (14 visits, 34 %), dropped by half on Day 2, briefly rebounded on Day 3, and tapered off to four visits on Day 5:

$$\text{Day totals} = \langle 14,\ 7,\ 10,\ 6,\ 4 \rangle$$

This pattern suggests initial curiosity followed by habituation; sustained reflection therefore requires additional prompts (e.g., weekly summaries or push reminders) rather than a purely on-demand dashboard. The usage pattern supports **H1** by illustrating that dashboard-based summary feedback temporarily raised awareness levels.

	Day 1	Day 2	Day 3	Day 4	Day 5	Total
User 1	1	0	2	0	1	4
User 2	1	2	0	3	0	6
User 3	1	0	1	0	0	2
User 4	0	1	2	0	0	3
User 5	0	1	2	0	0	3
User 6	1	0	1	0	2	4
User 7	0	0	0	0	0	0
User 8	7	2	1	2	0	12
User 9	0	0	0	1	0	1
User 10	3	1	1	0	1	6
User 11	0	0	0	0	0	0

Table 6.5: Metrics visits per user and day

6.3.4 Mode Comparison

The three-mode toggle distributed user traffic across models with varying energy demands. Table 6.6 compares the share of prompts, token counts, and energy consumption across modes.

Mode	Prompts	Tokens in	Tokens out	Energy
Energy-Efficient	155 (55.8%)	60473 (26.1%)	213350 (47.2%)	6.98Wh (4.1%)
Balanced	54 (19.4%)	83213 (35.9%)	82272 (18.2%)	11.51Wh (6.8%)
Performance	69 (24.8%)	88093 (38.0%)	156179 (34.6%)	149.55Wh (89.0%)
Total	278 (100%)	231779 (100%)	451801 (100%)	168.04Wh (100%)

Table 6.6: Share of prompts, tokens, and energy consumption per mode

Performance mode accounted for 24.8% of prompts but 89.0% of total energy usage. Energy-Efficient mode processed the majority of prompts (55.8%) but consumed only 4.1% of the total.

To control for prompt length, Table 6.7 compares each mode's energy consumption normalized by input token count only. This isolates energy per 1,000 input tokens and excludes any weighting of output. Additionally, the constant overhead of (0.020 Wh) has been excluded from the usage.

Performance mode consumed approximately 13.4 times more energy per input token than balanced mode. Balanced mode consumed about 1.95 times more than Energy-Efficient mode.

Mode	Tokens in	Total	Overhead	Token based	per 1k Tokens in
Energy-Efficient	60,473	6.98Wh	3.1Wh	3.88Wh	0.064Wh
Balanced	83,213	11.51Wh	1.08Wh	10.43Wh	0.125Wh
Performance	88,093	149.55Wh	1.38Wh	148.17Wh	1.682Wh

Table 6.7: Energy consumption per 1,000 input tokens by mode

Figure 6.1 illustrates the daily distribution of prompts per mode. Performance mode dominated on Day 1, then decreased during the week with a small spike on Day 5, while Energy-Efficient mode increased steadily, peaking on Day 4, with a small decrease on Day 5. This indicates that users adapted their behavior over the week. Balanced mode remained relatively stable until Day 3, then dropped to a lower level on Days 4 and 5.

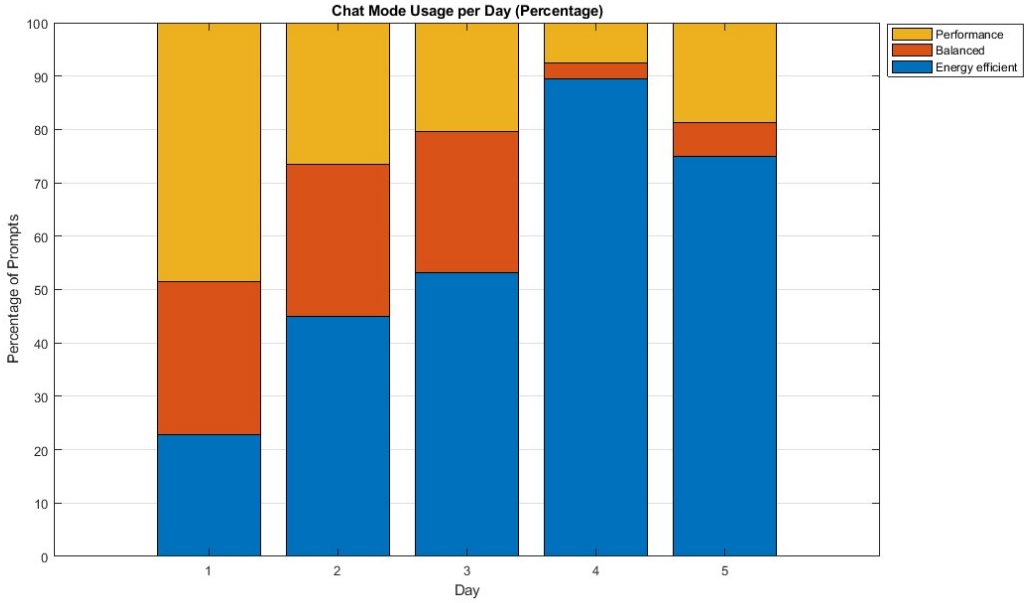


Figure 6.1: Total prompts sent per mode per day

6.3.5 User-Level Patterns

Three heavy users (IDs 8, 5, 3) generated **77%** of the study's total energy footprint as shown in Table 6.8; ID 8 alone accounted for nearly half of it, owing to a 64% reliance on Performance mode. Conversely, eight participants sent a majority of prompts in Energy-Efficient-mode, thus reaching a much lower total energy consumption. This supports **H2**, indicating that higher individual awareness is associated with energy-saving behavior, as seen in both mode choices and per-user footprints.

User	Mode	Prompts	%	In	Out	Usage (Wh)
1	Energy efficient	9	47.368	1513	7588	0.31383
	Balanced	8	42.105	1313	3694	0.56544
	Performance	2	10.526	272	2748	2.378
	Total	19	100	3098	14030	3.2572
2	Energy efficient	26	74.286	7297	37641	1.183
	Balanced	7	20	4407	8760	1.1341
	Performance	2	5.7143	2239	4150	3.9547
	Total	35	100	13943	50551	6.2718
3	Energy efficient	15	53.571	5298	27434	0.78314
	Balanced	0	0	0	0	0
	Performance	13	46.429	27227	27878	29.116
	Total	28	100	32525	55312	29.9
4	Energy efficient	4	57.143	166	5905	0.1799
	Balanced	3	42.857	264	1299	0.19759
	Performance	0	0	0	0	0
	Total	7	100	430	7204	0.37749
5	Energy efficient	9	31.034	12321	10920	0.4152
	Balanced	13	44.828	68652	36767	5.6961
	Performance	7	24.138	4850	15288	13.848
	Total	29	100	85823	62975	19.959
6	Energy efficient	17	60.714	3137	23811	0.7532
	Balanced	6	21.429	3941	6732	0.8979
	Performance	5	17.857	3800	11202	10.196
	Total	28	100	10878	41745	11.847
7	Energy efficient	0	0	0	0	0
	Balanced	4	100	671	4900	0.59083
	Performance	0	0	0	0	0
	Total	4	100	671	4900	0.59083
8	Energy efficient	10	18.868	7853	15612	0.49526
	Balanced	9	16.981	2753	14733	1.7345
	Performance	34	64.151	48302	79649	76.932
	Total	53	100	58908	109990	79.162
9	Energy efficient	65	98.485	22888	84439	2.8147
	Balanced	1	1.5152	5	48	0.024964
	Performance	0	0	0	0	0
	Total	66	100	22893	84487	2.8397
10	Energy efficient	0	0	0	0	0
	Balanced	0	0	0	0	0
	Performance	6	100	1403	15264	13.084
	Total	6	100	1403	15264	13.084
11	Energy efficient	0	0	0	0	0
	Balanced	3	100	1207	5339	0.62859
	Performance	0	0	0	0	0
	Total	3	100	1207	5339	0.62859

Table 6.8: Aggregated prompts per user and per mode

6.3.6 Token Length Observations

The average input length remained stable throughout the week, suggesting that participants did not adjust their prompts to reduce length over time. Participant comments indicated that "rewriting takes too much time," implying that selecting a different model was perceived as a more efficient strategy. Figure 6.2 shows that users rarely edited their input during typing, particularly with regard to shortening prompts in response to model predictions.

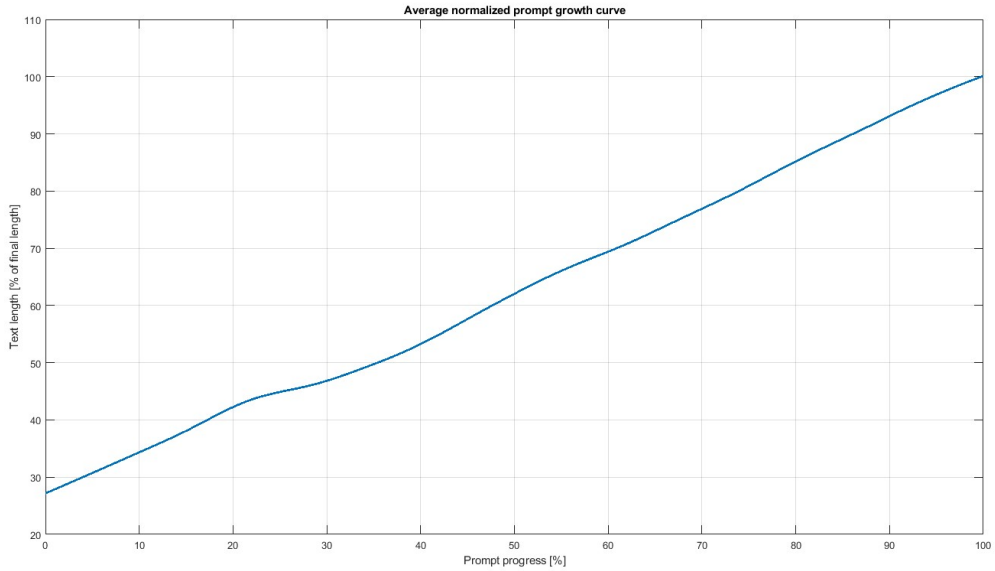


Figure 6.2: Average normalized prompt growth curve

Furthermore, Figure 6.3 offers an additional perspective by presenting a scatter plot of input versus output tokens (95th percentile), including both the output prediction function $f(x) = x$ defined in section 4.4.3 and a LOWESS [104] trend of the actual prediction.

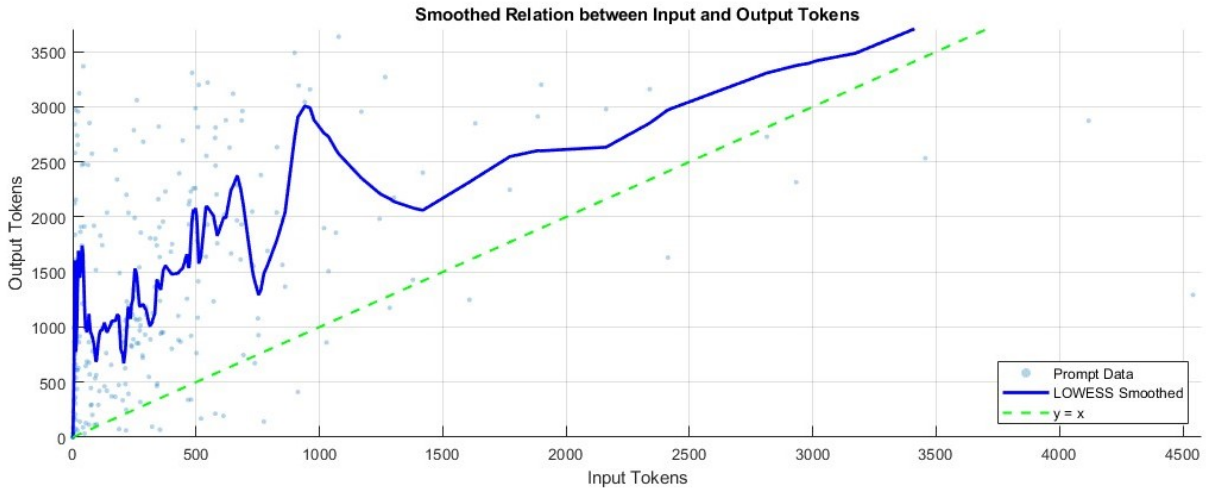


Figure 6.3: Scatter plot of input vs. output tokens with LOWESS trend and prediction function

The plot shows that the prediction function did not accurately estimate output tokens in most cases. On average, the number of output tokens was higher than the number of input tokens. The distribution of

prompts per conversation shows a mean of 4.03, a median of 3, and a mode of 1, indicating that while some users sent multiple prompts, many conversations consisted of only one.

6.4 Summary

6.4.1 Self-Reported Awareness and Behavior

Self-report ratings reveal the following patterns in participants awareness and energy-saving behavior:

- Awareness increased over the five days, peaking on Day 5 ($M = 4.44$), aligned with highest dashboard activity. **[Supports H1]**
- 81.8% of participants regularly used Energy-Efficient-mode when accuracy was not critical ($M = 4.18$). **[Supports H2]**
- Prompt shortening was rated neutrally ($M = 3.00$); log data confirmed no substantial input length reduction.
- All features scored above $4/5$ in usability, with the inline *mode toggle* rated highest ($M = 4.64$).
- Qualitative feedback highlighted needs for:
  - **T1:** Proactive push reminders,
  - **T2:** Real-time usage feedback in relatable units (e.g., *Laptop-minutes*),
  - **T3:** Periodic contextual framing or "wake-up calls".

6.4.2 Behavioral Log Findings

System logs show how each mode's use affected energy consumption and how participants choices shifted over time:

- Despite handling only 24.8% of prompts, **Performance mode** caused 89% of energy usage.
- **Energy-Efficient-mode** handled 55.8% of prompts with just 4.1% of the energy footprint.
- Energy consumption per 1,000 input tokens:
  - Energy-Efficient-mode: **0.064 Wh**
  - Balanced: **0.125 Wh** (1.95× more than Energy-Efficient)
  - Performance: **1.682 Wh** (13× more than Balanced)
- Energy-Efficient-mode usage increased over the week, indicating behavioral adaptation. **[Supports H2]**
- Dashboard usage was front-loaded (Day 1: 14 visits), then tapered off, suggesting that sustained reflection requires push-based reminders. **[Supports H1]**
- Three heavy users (IDs 8, 5, 3) caused 77% of total energy usage, with ID 8 alone responsible for 47%.
- Input lengths remained stable; participants favored mode selection over prompt rewriting due to time efficiency.

6.5 Hypothesis Validation Summary

H1. Showing per-prompt consumption and predicted consumption increases awareness.

Supported.

- Mean awareness increased from 3.27 (Day 1) to 4.44 (Day 5).
- Awareness peaks coincided with dashboard usage and high feature visibility.
- Inline and summary cues (Energy-Note, Mode-Toggle, Dashboard) visibly contributed to raising awareness.

H2. Awareness scores are positively correlated with pro-sustainability behavior.

Supported.

- Higher awareness levels (Days 2 and 5) aligned with increased Energy-Efficient-mode usage.
- Participants with higher self-reported awareness also demonstrated lower per-user energy footprints.
- Prompt length remained stable; mode-switching was the primary behavioral adjustment.

7 Discussion

This chapter reflects on the implications of the findings, evaluates the hypotheses in light of user behavior, addresses limitations of the study, and outlines directions for future work.

7.1 Interpretation and Evaluation

The experiment showed that even lightweight, non-intrusive UI changes can nudge users toward more energy-conscious decisions in LLM-based interactions. Awareness increased measurably, and most participants adopted energy-saving behaviors without being forced to. However, the extent of behavioral change varied widely between users. The results suggest that the success of such interventions depends not only on their design but also on user intent, attention, and usage style.

While mode switching was a low-effort action with high uptake, reducing prompt length appeared cognitively costly and was largely avoided. This asymmetry reveals an important insight: Users are more likely to adopt energy-efficient behavior when the intervention aligns with their workflow and requires minimal effort. In contrast, behaviors that demand extra cognitive investment like rewriting queries which may need stronger incentives or automation.

Moreover, usage concentration among a few participants highlights the "heavy-user bias": A small group can dominate total energy use regardless of system nudges. For these users, awareness alone may not be enough, additional strategies such as adaptive defaults or tailored recommendations may be needed.

7.2 Evaluation of Energy consumption

The total number of prompts and the overall energy consumption observed during the experiment were lower than initially expected. Although the underlying formula used for estimating consumption is largely based on empirical observations rather than disclosed usage data, it raises the valid question of whether the absolute energy consumption per user is truly significant.

When compared to familiar energy analogies such as gaming on a PlayStation 5 (168 Wh $\approx$ 48 minutes of playtime) or using a laptop (3.7 hours) the consumption appears relatively low and arguably negligible. However, this perspective shifts when considering the energy saved through users actively selecting the energy-efficient mode. This behavioral change highlights the potential of UI-based interventions to reduce energy usage.

If all interactions had occurred exclusively in Performance Mode, total energy consumption would have increased from approximately 168 Wh to an estimated 429 Wh, calculated as follows:

$$231{,}779 \cdot 0.00021 + 451{,}801 \cdot 0.00083 + 278 \cdot 0.020 \approx 429\,\text{Wh}$$

This reinforces the relevance of optimizing user interfaces for energy awareness. Moreover, as this experiment was limited to a single prototype, the actual energy footprint across other conversational AI tools exceeds the measured values, further underscoring the importance of sustainable design in AI applications.

7.3 Answering the Research Questions

RQ1: To what extent are users currently aware of the energy implications associated with their chatbot interactions?

Initial awareness was moderate and grew over time, confirming that awareness is improvable via simple interventions. Participants responded well to inline cues and dashboard summaries. Still, awareness fluctuated depending on salience and novelty, suggesting that reminders need to be periodically refreshed.

RQ2: How can UI-based features most effectively increase user awareness regarding the energy consumption of conversational AI?

Features were most effective when integrated seamlessly into the workflow. The inline toggle and energy note stood out due to constant visibility and clarity. In contrast, less frequently accessed features like the dashboard had high value but required user initiative. Real-time, ambient cues appear more sustainable than static overviews.

RQ3: How strongly does increased user awareness correlate with reductions in conversational AI energy consumption?

Awareness was linked to energy-saving behavior, particularly through mode selection. However, this correlation was dampened by individual differences: Some users consistently optimized their usage, while others reverted to high-consumption modes for complex tasks. Prompt length remained largely unaffected. The relationship is therefore present but not uniform.

7.4 Limitations

Although the research questions could be answered and the hypothesis were at least partially supported this results should be seen in light of the constraints and limitations of this project:

- **Sample size and bias:** With only 11 participants, mostly with technical backgrounds, the sample is too narrow for generalization.
- **Energy approximation:** Due to lack of public data from OpenAI, energy usage was estimated via heuristic models. This introduces uncertainty in all consumption metrics.
- **Output prediction:** The output token length estimation heuristic proved to be an imprecise predictor, especially for longer prompts. This affects the accuracy of energy feedback and weakens user trust in numerical feedback. More sophisticated models like regression or transformer-specific predictors are needed for future iterations. This might also be a factor why participants did not change their prompt length.
- **Self-selection bias:** Participants were self-motivated and likely more aware of sustainability issues, which may not reflect the general population.
- **Limited feature set:** The prototype focused on a few UI changes; other potential interventions (e.g., features discussed in the conceptual solution in the decision matrix Table 4.4) were not tested.
- **Short duration:** Five days are sufficient for observing short-term change but not long-term habits or retention.
- **Limited granularity:** Some behaviors (e.g., mental effort, editing hesitations) are not captured in the logs, leaving gaps in interpretation.

7.5 Future Work and Development

7.5.1 Research and Validation

This thesis has demonstrated both the potential and the necessity for more extensive research in this emerging field. Several aspects remain insufficiently explored, and notable blind spots persist. To address these gaps, future research should:

- Incorporate real energy data from API providers to validate energy estimations.
- Expand the participant pool with non-technical and more diverse user groups.
- Run longitudinal studies to investigate lasting behavioral change and intervention fatigue.
- Introduce richer behavior tracking to detect prompt refinement, hesitation, or adaptation patterns.

7.5.2 Prototype Improvements

While the current prototype successfully demonstrated the feasibility of UI-based energy awareness features, several enhancements could further increase its effectiveness, usability, and adaptability. Future iterations should aim to:

- Add user-configurable modes or AI-guided suggestions based on usage patterns.
- Improve energy and output predictions with data-driven models.
- Implement features proposed in Section 4, such as real-time "current usage" displays and periodic "wake-up calls."
- Add detection of trivial prompts and recommend simpler alternatives like a Translation Service or, calculator etc.

7.5.3 User Guidance and Framing

The experiment revealed that participants responded positively to clear and relatable feedback, especially when energy consumption was contextualized through comparisons (e.g., charging a smartphone). Building on this insight, future designs could enhance user engagement and awareness through:

- Emotional and contextual framing ("this equals powering a fridge for X hours").
- Push reminders with usage summaries or tailored advice.
- More interactive dashboards showing impact over time.

7.6 Concluding Remarks

UI-only strategies show real potential for raising awareness and enabling more sustainable behavior in conversational AI interfaces. However, their impact depends on user context, feature salience and the perceived effort–benefit ratio. Future work should aim to make these interventions adaptive, personalized and grounded in real data.

8 Conclusion

This thesis explored the potential of UI-only strategies to increase awareness and reduce energy consumption in conversational AI usage. Motivated by the growing inference-related energy demand of large language models, we investigated whether lightweight frontend interventions could measurably influence user behavior without compromising usability.

The findings of our five-day field experiment with frequent LLM users indicate that lightweight frontend interventions can have a measurable impact. Participants showed a significant increase in awareness (M = 4.44/5), and those with higher awareness scores consistently opted for more energy-efficient settings. More than half of all prompts were routed through the energy-saving mode and the overall usage resulted in an estimated 35% reduction in energy consumption compared to a performance-mode baseline. In particular, these reductions were achieved without negatively affecting user experience or perceived output quality, suggesting that sustainability and usability can coexist in the same interface.

Beyond the empirical results, this work contributes to a growing discourse within Sustainable Human-Computer Interaction. While prior research has highlighted the perceptual, technical, and design gaps that hinder energy-conscious digital behavior, our work demonstrates a concrete, scalable way to address them through intentional interface design. The study also reinforces the idea that users are not indifferent to energy concerns; instead, they often lack accessible, interpretable feedback to guide their decisions. By bridging this awareness-action gap, UI interventions can act as effective nudges in everyday systems.

However, several limitations must be acknowledged. Our study involved a relatively small technically literate user group, and energy estimates were derived using an approximation model based on token counts and pricing-based coefficients. While this approach is sufficient for behavioral nudging, it may not reflect actual backend power consumption with high accuracy. Furthermore, long-term behavioral retention and broader user demographics remain unexplored. These factors limit generalisability and should be addressed in future work.

Looking ahead, this thesis opens the door for further research into sustainability-aware design strategies for AI-powered applications. Opportunities include integrating live energy telemetry from commercial providers, exploring adaptive interfaces that respond dynamically to carbon intensity, and embedding these ideas in educational or workplace contexts.

Ultimately, as AI systems become embedded in the daily workflows of millions, energy use will no longer be just a backend concern, it will be a user-facing issue. By giving users the tools and information to make responsible choices, interface designers have a critical role to play in shaping a more sustainable digital future. This thesis has shown that even modest UI changes can lead to meaningful environmental benefits, setting the stage for a new kind of interaction design: One that is not only human-centered, but energy-aware.

References

[1] W. Vanderbauwhede, *Estimating the increase in emissions caused by ai-augmented search*, 2025. arXiv: 2407.16894 [cs.CY]. [Online]. Available: https://arxiv.org/abs/2407.16894.

[2] A. de Vries, „The growing energy footprint of artificial intelligence“, *Joule*, vol. 7, no. 10, pp. 2191–2194, Oct. 18, 2023, Publisher: Elsevier, ISSN: 2542-4785. DOI: 10.1016/j.joule.2023.09.004. [Online]. Available: https://doi.org/10.1016/j.joule.2023.09.004 (visited on Jul. 14, 2025).

[3] P. Jiang, C. Sonne, W. Li, F. You, and S. You, „Preventing the immense increase in the life-cycle energy and carbon footprints of llm-powered intelligent chatbots“, *Engineering*, vol. 40, pp. 202–210, Sep. 2024. DOI: 10.1016/j.eng.2024.04.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2095809924002315.

[4] E. Strubell, A. Ganesh, and A. McCallum, „Energy and policy considerations for deep learning in nlp“, in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 3645–3650. [Online]. Available: https://arxiv.org/abs/1906.02243.

[5] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, *Towards the systematic reporting of the energy and carbon footprints of machine learning*, 2022. arXiv: 2002.05651 [cs.CY]. [Online]. Available: https://arxiv.org/abs/2002.05651.

[6] X. Wang, C. Na, E. Strubell, S. Friedler, and S. Luccioni, „Energy and carbon considerations of fine-tuning bert“, in *Findings of the Association for Computational Linguistics: EMNLP 2023*, Association for Computational Linguistics, 2023, pp. 9058–9069. DOI: 10.18653/v1/2023.findings-emnlp.607. [Online]. Available: http://dx.doi.org/10.18653/v1/2023.findings-emnlp.607.

[7] B. Capital, „Trends in artificial intelligence“, Bond Capital, Tech. Rep., 2025, ChatGPT reached 800 million weekly active users in 17months. [Online]. Available: https://www.bondcap.com/report/pdf/Trends_Artificial_Intelligence.pdf.

[8] C. Preist, D. Schien, and E. Blevis, „Understanding and mitigating the effects of device and cloud service design decisions on the environmental footprint of digital infrastructure“, in *Proceedings of the 34th Annual ACM Conference on Human Factors in Computing Systems (CHI'16)*, CHI 2016: Conference on Human Factors in Computing Systems, San Jose, CA, USA, Association for Computing Machinery, May 2016, pp. 1324–1337, ISBN: 9781450333627. DOI: 10.1145/2858036.2858378. [Online]. Available: https://doi.org/10.1145/2858036.2858378.

[9] S. Poddar, P. Koley, J. Misra, S. Podder, N. Ganguly, and S. Ghosh, *Towards sustainable nlp: Insights from benchmarking inference energy in large language models*, 2025. arXiv: 2502.05610 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2502.05610.

[10] J. Fernandez, C. Na, V. Tiwari, Y. Bisk, S. Luccioni, and E. Strubell, *Energy considerations of large language model inference and efficiency optimizations*, 2025. arXiv: 2504.17674 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2504.17674.

[11] B. Li, Y. Jiang, V. Gadepally, and D. Tiwari, *Toward sustainable genai using generation directives for carbon-friendly large language model inference*, 2024. arXiv: 2403.12900 [cs.DC]. [Online]. Available: https://arxiv.org/abs/2403.12900.

[12] J. Stojkovic, C. Zhang, Í. Goiri, J. Torrellas, and E. Choukse, *Dynamollm: Designing llm inference clusters for performance and energy efficiency*, 2024. arXiv: 2408.00741 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2408.00741.

[13] D. Geelen, R. Mugge, S. Silvester, and A. Bulters, „The use of apps to promote energy saving: A study of smart-meter-related feedback in the netherlands“, *Energy Efficiency*, vol. 12, no. 6, pp. 1635–1660, 2019. DOI: 10.1007/s12053-019-09777-z. [Online]. Available: https://doi.org/10.1007/s12053-019-09777-z.

[14] S. Darby, „The effectiveness of feedback on energy consumption“, *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, vol. 486, no. 3, pp. 93–109, Jan. 2006, immediate feedback =¿ +5% savings. [Online]. Available: https://smartgridawareness.org/wp-content/uploads/2016/05/effectiveness-of-feedback-on-energy-consumption-darby-2006.pdf.

[15] R. Likert, „A technique for the measurement of attitudes“, *Archives of Psychology*, vol. 22, no. 140, pp. 1–55, 1932.

[16] B. Kitchenham, S. L. Pfleeger, and L. Madeyski, „Guidelines for performing experiments in software engineering“, Keele University & Durham University, Joint Technical Report, Staffordshire, UK, Tech. Rep. EBSE-TR-2013-003, 2013, Version 2.0, updated January 2013. [Online]. Available: https://www.ebse.org.uk/.

[17] J. Walters, A. Nair, P. Mastronardi, A. Li, and X. Bai, „Purple: Combining individual and collective action to increase online sustainability“, in *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, ser. CHI EA 25, New York, NY, USA: Association for Computing Machinery, 2025, ISBN: 9798400713958. DOI: 10.1145/3706599.3720304. [Online]. Available: https://doi.org/10.1145/3706599.3720304.

[18] J. Chen, G. Fu, Z. Ren, M. Li, and J. Ham, „Effects of anthropomorphic design cues of chatbots on users' perception and visual behaviors“, *International Journal of Human–Computer Interaction*, vol. 40, no. 14, pp. 3636–3654, Jul. 2024. DOI: 10.1080/10447318.2023.2193514. [Online]. Available: https://doi.org/10.1080/10447318.2023.2193514.

[19] J. Kelly and W. Knottenbelt, *Does disaggregated electricity feedback reduce domestic electricity consumption? a systematic review of the literature*, 2016. arXiv: 1605.00962 [cs.CY]. [Online]. Available: https://arxiv.org/abs/1605.00962.

[20] H. Allcott and T. Rogers, „The short-run and long-run effects of behavioral interventions: Experimental evidence from energy conservation“, *American Economic Review*, vol. 104, no. 10, pp. 3003–37, Oct. 2014. DOI: 10.1257/aer.104.10.3003. [Online]. Available: https://www.aeaweb.org/articles?id=10.1257/aer.104.10.3003.

[21] S. LeVine, *How peer pressure can help save the planet*, https://www.axios.com/2019/01/30/harvard-business-review-peer-pressure-energy-conservation, Accessed: 2025-07-06, 2019.

[22] B. Huseynli, „Gamification in energy consumption: A model for consumers' energy saving“, *International Journal of Energy Economics and Policy*, vol. 14, no. 1, pp. 312–320, 2024, proposes gamified energy-saving model. [Online]. Available: http://dx.doi.org/10.32479/ijeep.14395.

[23] Epoch AI. „How much energy does chatgpt use?“ (Feb. 10, 2025), [Online]. Available: https://epoch.ai/gradient-updates/how-much-energy-does-chatgpt-use (visited on Jun. 11, 2025).

[24] S. Altman, *The gentle singularity*, https://blog.samaltman.com/the-gentle-singularity, 2025.

[25] E. Masanet, A. Shehabi, N. Lei, S. Smith, and J. Koomey, „Recalibrating global data center energy-use estimates“, *Environmental Research Letters*, vol. 13, no. 1, p. 014 003, 2018, Relevant for recalculating per-search energy based on efficiency gains. DOI: 10.1088/1748-9326/aa9671. [Online]. Available: https://datacenters.lbl.gov/sites/default/files/Masanet_et_al_Science_2020.full_.pdf.

[26] AppleInsider, *Apple's iphone 14 battery capacities revealed in filing*, 2022.

[27] N. Kumar. „65 chatbot statistics for 2025 — new data released“. DemandSage. (Jan. 28, 2025), [Online]. Available: https://www.demandsage.com/chatbot-statistics/ (visited on Mar. 17, 2025).

[28] S. Samsi, D. Zhao, J. McDonald, *et al.*, *From words to watts: Benchmarking the energy costs of large language model inference*, arXiv preprint 2310.03003, 2023. DOI: 10.48550/arXiv.2310.03003. [Online]. Available: https://arxiv.org/abs/2310.03003.

[29] B. Li, Y. Jiang, V. Gadepally, and D. Tiwari, „Sprout: Green generative AI with carbon-efficient LLM inference“, in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 21 799–21 813. DOI: 10.18653/v1/2024.emnlp-main.1215. [Online]. Available: https://aclanthology.org/2024.emnlp-main.1215/.

[30] T. Shi, Y. Wu, S. Liu, and Y. Ding, *Greenllm: Disaggregating large language model serving on heterogeneous gpus for lower carbon emissions*, 2024. arXiv: 2412.20322 [cs.AR]. [Online]. Available: https://arxiv.org/abs/2412.20322.

[31] E. J. Husom, A. Goknil, L. K. Shar, and S. Sen, *The price of prompting: Profiling energy use in large language models inference*, 2024. arXiv: 2407.16893 [cs.CY]. [Online]. Available: https://arxiv.org/abs/2407.16893.

[32] Z. Fu, F. Chen, S. Zhou, H. Li, and L. Jiang, *Llmco2: Advancing accurate carbon footprint prediction for llm inferences*, 2024. arXiv: 2410.02950 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2410.02950.

[33] M. Argerich and M. Patiño-Martínez, „Measuring and improving the energy efficiency of large language models inference“, *IEEE Access*, vol. PP, pp. 1–1, Jan. 2024. DOI: 10.1109/ACCESS.2024.3409745.

[34] R. Rubei, A. Moussaid, C. di Sipio, and D. di Ruscio, *Prompt engineering and its implications on the energy consumption of large language models*, 2025. arXiv: 2501.05899 [cs.SE]. [Online]. Available: https://arxiv.org/abs/2501.05899.

[35] A. Nik, M. A. Riegler, and P. Halvorsen, „Energy-conscious llm decoding: Impact of text generation strategies on gpu energy consumption“, 2025. DOI: 10.48550/arXiv.2502.11723. arXiv: 2502.11723 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2502.11723.

[36] G. Wilkins, S. Keshav, and R. Mortier, *Offline energy-optimal llm serving: Workload-based energy models for llm inference on heterogeneous systems*, 2024. arXiv: 2407.04014 [cs.DC]. [Online]. Available: https://arxiv.org/abs/2407.04014.

[37] G. Wilkins, S. Keshav, and R. Mortier, *Hybrid heterogeneous clusters can lower the energy consumption of llm inference workloads*, 2024. arXiv: 2407.00010 [cs.DC]. [Online]. Available: https://arxiv.org/abs/2407.00010.

[38] M. Adamska, D. Smirnova, H. Nasiri, Z. Yu, and P. Garraghan, *Green prompting*, 2025. arXiv: 2503.10666 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2503.10666.

[39] L. Solovyeva, S. Weidmann, and F. Castor, *Ai-powered, but power-hungry? energy efficiency of llm-generated code*, 2025. DOI: 10.48550/arXiv.2502.02412. arXiv: 2502.02412 [cs.SE]. [Online]. Available: https://arxiv.org/abs/2502.02412.

[40] T. Coignion, C. Quinton, and R. Rouvoy, *Green my llm: Studying the key factors affecting the energy consumption of code assistants*, 2024. DOI: 10.48550/arXiv.2411.11892. arXiv: 2411.11892 [cs.SE]. [Online]. Available: https://arxiv.org/abs/2411.11892.

[41] V. Nguyen, H. Huynh, V. Dhopate, *et al.*, „On-device or remote? on the energy efficiency of fetching llm-generated content“, in *CAIN 2025: Architecting and Testing AI Systems*, Empirical comparison showing remote server inference uses 4–9× less client energy than on-device LLMs, 2025.

[42] A. Isaza-Giraldo, P. Bala, P. Campos, and L. Pereira, „Prompt-gaming: A pilot study on llm-evaluating agent in a meaningful energy game“, May 2024, pp. 1–12. DOI: 10.1145/3613905.3650774.

[43] L. Anthony, B. Kanding, and R. Selvan, „Carbontracker: Tracking and predicting the carbon footprint of training deep learning models“, in *Proceedings of the ICML 2020 Workshop on Energy Efficient ML*, 2020. [Online]. Available: https://arxiv.org/abs/2007.03051.

[44] B. Courty, V. Schmidt, S. Luccioni, *et al.*, *Mlco2/codecarbon: V2.4.1*, version v2.4.1, May 2024. DOI: 10.5281/zenodo.11171501. [Online]. Available: https://doi.org/10.5281/zenodo.11171501.

[45] ScaleDown Team, *Scaledown, Ai productivity & sustainability suite — chrome extension*, Browser extension, version 0.2.4, Available on the Chrome Web Store, Apr. 29, 2025. [Online]. Available: https://chromewebstore.google.com/detail/scaledown/jofapkamgblhjajlppnaiomcjhnllhnd (visited on Jun. 11, 2025).

[46] Pascal, *Ai wattch, Track chatgpt's carbon emissions — chrome extension*, Browser extension, version 1.5, Available on the Chrome Web Store, Jun. 2, 2025. [Online]. Available: https://chromewebstore.google.com/detail/ai-wattch-%E2%80%93-track-chatgpt/meacendfnhnjbkmfbfogbmekkhnamffn (visited on Jun. 11, 2025).

[47] ScaleDown Team. „Methodology & accuracy notes“. Accessed 11 Jun 2025. (), [Online]. Available: https://scaledown.ai/methodology (visited on Jun. 11, 2025).

[48] S. K. Dam, C. S. Hong, Y. Qiao, and C. Zhang, *A complete survey on llm-based ai chatbots*, 2024. arXiv: 2406.16937 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2406.16937.

[49] comparis.ch AG, *Medienmitteilung: Zwei drittel der schweizerinnen und schweizer haben schon chatgpt oder gemini genutzt*, Presseportal, Zürich, 18. März 2025, Mar. 2025. [Online]. Available: https://www.presseportal.ch/de/pm/100003671/100929653.

[50] L. F. W. Anthony, B. Kanding, and R. Selvan, *Carbontracker: Tracking and predicting the carbon footprint of training deep learning models*, 2020. arXiv: 2007.03051 [cs.CY]. [Online]. Available: https://arxiv.org/abs/2007.03051.

[51] T. Rist and M. Masoodian, „Promoting sustainable energy consumption behavior through interactive data visualizations“, *Multimodal Technologies and Interaction*, vol. 3, no. 3, 2019, ISSN: 2414-4088. DOI: 10.3390/mti3030056. [Online]. Available: https://www.mdpi.com/2414-4088/3/3/56.

[52] CHEMTREC®. „Apis & bpis report (updated 09 sep 2022)“. Battery capacity reference used for iPhone 14. (2022), [Online]. Available: https://app.clickdimensions.com/blob/chemtreccom-ajrb1/files/apis_bpisreportupdated20220909.pdf (visited on Jul. 1, 2025).

[53] Coolblue. „Stromverbrauch kühlschrank: Wie viel watt braucht ein kühlschrank?“ (2023), [Online]. Available: https://www.coolblue.de/beratung/verbrauch-kuehlschrank.html (visited on Jul. 1, 2025).

[54] U. Blog. „Stromverbrauch von laptops: Wie viel watt verbraucht ein laptop?“ (2024), [Online]. Available: https://de.ugreen.com/blogs/powerstation/stromverbrauch-von-laptops-wie-viel-watt-verbraucht-ein-laptop (visited on Jul. 1, 2025).

[55] Lampen&Leuchten. „Nachhaltigkeit und umwelt: Alles über led-beleuchtung“. (2024), [Online]. Available: https://www.lampenundleuchten.ch/c/beratung/alles-uber-led-beleuchtung/nachhaltigkeit-und-umwelt (visited on Jul. 1, 2025).

[56] Verivox. „Ps5: So hoch ist der stromverbrauch“. (2024), [Online]. Available: https://www.verivox.de/strom/ratgeber/ps5-so-hoch-ist-der-stromverbrauch-1120593/ (visited on Jul. 1, 2025).

[57] N. Jegham, M. Abdelatti, L. Elmoubarki, and A. Hendawi, „How hungry is ai? benchmarking energy, water, and carbon footprint of llm inference“, 2025. arXiv: 2505.09598 [cs.CY]. [Online]. Available: https://arxiv.org/abs/2505.09598.

[58] OpenAI, *Openai api pricing and tokenization*, https://openai.com/pricing, 2023.

[59] EpochAI, *Energy efficiency benchmarking of llms: Gpt-4, claude, gemini, and more*, https://epochai.org/blog/energy-benchmarking-llms, Accessed: 2025-07-01, 2024.

[60] OpenAI, *Tokenizer documentation*, https://platform.openai.com/tokenizer, 2023.

[61] M. Levy, A. Jacoby, and Y. Goldberg, *Same task, more tokens: The impact of input length on the reasoning performance of large language models*, 2024. arXiv: 2402.14848 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2402.14848.

[62] Microsoft, *Cloud native apps on azure*, https://learn.microsoft.com/en-us/dotnet/architecture/cloud-native/, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[63] GitHub, *Github actions for ci/cd*, https://docs.github.com/en/actions/automating-builds-and-tests/about-continuous-integration, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[64] Microsoft Corporation, *Auswählen zwischen herkömmlichen webanwendungen und single-page-webanwendungen (spas)*, https://learn.microsoft.com/de-de/dotnet/architecture/modern-web-apps-azure/choose-between-traditional-web-and-single-page-apps, Accessed: 2025-07-06, Jun. 2023.

[65] Microsoft, *Host a static website in azure storage*, https://learn.microsoft.com/en-us/azure/storage/blobs/storage-blob-static-website, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[66] Microsoft, *Azure functions overview*, https://learn.microsoft.com/en-us/azure/azure-functions/functions-overview, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[67] OpenAI, *Openai api documentation*, https://platform.openai.com/docs/introduction, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[68] Microsoft, *Microsoft entra id documentation*, https://learn.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-whatis, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[69] Microsoft, *Monitor azure functions with application insights*, https://learn.microsoft.com/en-us/azure/azure-functions/functions-monitoring, Zuletzt abgerufen am 28.06.2025. (visited on Jun. 28, 2025).

[70] Angular Team, *Angular v19 overview*, https://v19.angular.dev/overview, Zugriff am 5. Juli 2025, 2025.

[71] M. Maheshwari and S. Agarwal, „Energy consumption analysis of display technologies on mobile devices“, *International Journal of Computer Applications*, vol. 176, no. 35, pp. 6–11, 2020. DOI: 10.5120/ijca2020919984.

[72] Microsoft Docs. „Authentication and authorization in azure static web apps“. Reference for /.auth/me, /.auth/login/aad, and /.auth/logout. (2024), [Online]. Available: https://learn.microsoft.com/en-us/azure/static-web-apps/authentication-authorization (visited on Jul. 1, 2025).

[73] A. Team, *@angular/core*, https://www.npmjs.com/package/@angular/core, Version 17 (aktuell bei Abfrage), 2025. (visited on Jun. 28, 2025).

[74] A. Team, *@angular/material*, https://www.npmjs.com/package/@angular/material, Material Design components for Angular, 2025. (visited on Jun. 28, 2025).

[75] C. Contributors, *Chart.js*, https://www.npmjs.com/package/chart.js, Simple yet flexible JavaScript charting library, 2025. (visited on Jun. 28, 2025).

[76] M. P. Contributors, *Marked*, https://www.npmjs.com/package/marked, A markdown parser and compiler, 2025. (visited on Jun. 28, 2025).

[77] R. Contributors, *Rxjs*, https://www.npmjs.com/package/rxjs, Reactive Extensions Library for JavaScript, 2025. (visited on Jun. 28, 2025).

[78] Microsoft Learn, *Common web application architectures*, Accessed: 2025-07-09, 2023. [Online]. Available: https://learn.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures.

[79] Microsoft Learn, *Unit testing best practices*, Accessed: 2025-07-09, 2023. [Online]. Available: https://learn.microsoft.com/en-us/dotnet/core/testing/unit-testing-best-practices.

[80] OpenAI. „Chat - streaming“. Zuletzt abgerufen am 28.06.2025. (2024), [Online]. Available: https://platform.openai.com/docs/api-reference/chat-streaming (visited on Jun. 28, 2025).

[81] Mozilla Developer Network. „Server-sent events - web apis — mdn“. Zuletzt abgerufen am 28.06.2025. (2024), [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/Server-sent_events (visited on Jun. 28, 2025).

[82] Microsoft. „Configure azure functions app settings“. Zuletzt abgerufen am 28.06.2025. (2024), [Online]. Available: https://learn.microsoft.com/en-us/azure/azure-functions/functions-how-to-use-azure-function-app-settings (visited on Jun. 28, 2025).

[83] Microsoft, *Databases, containers, and items – azure cosmos db*, https://learn.microsoft.com/azure/cosmos-db/resource-model, 2024.

[84] Microsoft, *Authentication and authorization in azure static web apps*, https://learn.microsoft.com/en-us/azure/static-web-apps/authentication-authorization, 2024.

[85] Microsoft, *Manage roles and access control in azure static web apps*, https://learn.microsoft.com/en-us/azure/static-web-apps/authentication-authorization?tabs=github-actions#roles, 2024.

[86] M. Fowler, *Unit test*, https://martinfowler.com/bliki/UnitTest.html, 2024.

[87] Microsoft, *Azure functions - unit testing guidance*, https://learn.microsoft.com/en-us/azure/azure-functions/functions-test-a-function, 2024.

[88] Angular, *Testing angular applications*, https://angular.dev/guide/testing, 2024.

[89] Jasmine Team, *Jasmine – behavior-driven javascript*, Accessed: 2025-07-31, 2025. [Online]. Available: https://jasmine.github.io/.

[90] Karma Team, *Karma: Test runner for javascript*, https://www.npmjs.com/package/karma, version 6.4.4, Last published a year ago; accessed on 2025-07-31, 2025.

[91] Microsoft, *Azure static web apps documentation*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/static-web-apps/overview.

[92] Microsoft, *Azure front door documentation*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview.

[93] Microsoft, *Azure functions scale and hosting options*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/azure-functions/functions-scale.

[94] Microsoft, *Azure functions limits*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/azure-functions/functions-scale#limits.

[95] Microsoft, *Globally distributed applications with azure cosmos db*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/cosmos-db/global-distribution.

[96] Microsoft, *Autoscale provisioned throughput in azure cosmos db*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/cosmos-db/provisioned-throughput-autoscale.

[97] OpenAI, *Openai api documentation*, Accessed: 2025-07-05, 2024. [Online]. Available: https://platform.openai.com/docs/guides/completions.

[98] OpenAI, *How openai scales chatgpt*, Accessed: 2025-07-05, 2024. [Online]. Available: https://openai.com/blog/scaling-chatgpt.

[99] Microsoft, *Resiliency patterns and best practices in azure*, Accessed: 2025-07-05, 2024. [Online]. Available: https://learn.microsoft.com/en-us/azure/architecture/resiliency/.

[100] Microsoft, *Build and deploy a static web app using github actions*, https://learn.microsoft.com/en-us/azure/static-web-apps/deploy-github-actions, 2024.

[101] Microsoft, *Azure functions github action*, https://github.com/Azure/functions-action, 2024.

[102] Microsoft, *Infrastructure as code on azure*, https://learn.microsoft.com/en-us/azure/devops/learn/devops-at-microsoft/infrastructure-as-code, 2024.

[103] Microsoft, *Iac strategy, best practices, and tools*, https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/ready/azure-best-practices/infra-as-code/, 2024.

[104] MathWorks, *Lowess Smoothing – MATLAB & Simulink*, https://www.mathworks.com/help/curvefit/lowess-smoothing.html, Accessed: 2025-07-12, MathWorks, 2025.

Declaration of honesty

I hereby declare that any individual/pair/team- work submitted for assessment is entirely the product of my own, my partner's and my team's effort that we have correctly cited all text passages that do not originate from us, in accordance with standard academic citation rules (e.g. IEEE), and that we have clearly mentioned all sources used; that we have declared in the table Table 8.1 all aids used; that we have acquired all intangible rights to any materials we may have used, such as images or graphics, or that these materials were created by us; that the topic, the thesis or parts of it have not been used in an assessment of another module, unless this has been expressly agreed with the lecturer in advance and is stated as such; that we are aware that our work may be checked for plagiarism and for third-party authorship of human or technical origin; that we are aware that the FHNW School of Engineering and Environment (FHNW School of Computer Science) will pursue a violation of this declaration of authenticity and that disciplinary consequences may result from this.

Tool	**Usage**
AI-Chatbots(ChatGPT, Gemini)	Creativity & Brainstorming assistance and paraphrasing
Copilot & Codex	Coding assistance
MatLab	Generating graphics and statistical analysis
Microsoft Excel	Statistical analysis and cleaning data
Overleaf	Writing thesis and support with LaTeX
Visual Studio Code	Developing web application

Table 8.1: External tools used during thesis work

Windisch, 31. Juli 2025

Name: Jack Gläser

Signature: 

Name: Simon Lüscher

Signature: 

A Appendix

A.1 Survey Form and Results

Conversational AI & Energy: How Much Do You Know?

Conversational AI and LLM chatbots are everywhere, making life easier and conversations smarter. But have you ever wondered about the energy they consume? Every chat, every request, and every response requires computing power—and that means real-world energy use.

Did you know that with just 5 prompts you could power a 1 W device like a small speaker for 1 hour 💡

That makes us curious.

How aware are you of conversational AI's resource consumption? This short survey will take **less than five minutes**, and your insights will help us understand public awareness about conversational AI's environmental impact.

This survey is a part of the research project "**Designing towards higher user awareness: UI strategies for reducing conversational AI energy consumption**" at the School of Computer Science (HSI) of the University of Applied Sciences and Arts Northwestern Switzerland (FHNW).

Our aim is to better understand how people use conversational AI or LLM chatbots, shed light on their environmental impact, and discover what features might help to use them more sustainably. Therefore, your voice matters!

The survey will be open for a month, ***from 18.03.2025 to 18.04.2025.***

You can complete it anonymously; if you'd like to see the results, please leave your email address at the end.

Note that we will use the terms **"conversational AI"** and **"LLM chatbot"** interchangeably in this survey.

👉 Start the survey now!

* Gibt eine erforderliche Frage an

Usage Patterns - Part 1

First, tell us about your experience and habits when using conversational AI tools.

1. How would you describe your overall technical proficiency? *

   Markieren Sie nur ein Oval.

   - ◯ Beginner (little to no technical knowledge)
   - ◯ Intermediate (comfortable using the technology for basic tasks)
   - ◯ Advanced (can code or work with AI tools)

2. Which of the following best describes your familiarity with conversational AI (e.g., ChatGPT, Bard, Claude, etc.)? *

   Markieren Sie nur ein Oval.

   - ◯ I've never heard of them (no familiarity) *Fahren Sie mit Frage 11 fort*
   - ◯ I've heard of them but never used them (aware, but not a user) *Fahren Sie mit Frage 11 fort*
   - ◯ I've used them occasionally (infrequent user) *Fahren Sie mit Frage 3 fort*
   - ◯ I'm a regular user (frequent user) *Fahren Sie mit Frage 3 fort*
   - ◯ I'm an advanced user (heavy/integrated usage) *Fahren Sie mit Frage 3 fort*

Usage Patterns - Part 2

Now, we would like to get to know your conversational AI usage in more detail.

3. How frequently do you use conversational AI tools? *

   Markieren Sie nur ein Oval.

   - ◯ Several times a day
   - ◯ A few times a week
   - ◯ A few times a month
   - ◯ Once a few months

4. What types of conversational AI services do you use? (please select all that apply) *

Wählen Sie alle zutreffenden Antworten aus.

- [ ] Text-based AI (e.g., ChatGPT, Google Bard)
- [ ] Image / Video generation AI (e.g., DALL·E, MidJourney, Runway)
- [ ] Code assistance (e.g., GitHub Copilot, Tabnine)
- [ ] AI-driven search engines (e.g., Perplexity, Google Search AI)
- [ ] Speech recognition/assistants (e.g., Siri, Alexa, Google Assistant)
- [ ] Sonstiges: _____

5. How do you **mainly** work with LLMs? *

Markieren Sie nur ein Oval.

- ◯ I use tools, like ChatGPT, Claude etc directly
- ◯ I directly interact with the LLMs via API or custom integrations
- ◯ Sonstiges: _____

6. What are your primary reasons for using LLM chatbots? (please select all that apply) *

Wählen Sie alle zutreffenden Antworten aus.

- [ ] Searching for information
- [ ] Research & learning
- [ ] Creative writing/content generation
- [ ] Coding assistance
- [ ] Entertainment & casual conversations
- [ ] Personal organization (e.g., summarizing, scheduling)
- [ ] Sonstiges: _____

7. Roughly how much time do you spend **actively** using such tools or interacting with LLMs **on a typical day**? *

Markieren Sie nur ein Oval.

- ◯ Less than half an hour
- ◯ Half - One hour
- ◯ 1 - 2 hours
- ◯ More than 2 hours
- ◯ I am not sure how to measure my usage time
- ◯ I don’t see the necessity to track my time spent

8. Which device do you **primarily** use to access LLM chatbots? *

Markieren Sie nur ein Oval.

- ◯ Desktop/Laptop
- ◯ Smartphone
- ◯ Tablet

9. Do you use a paid license for access to an LLM chatbot (e.g., ChatGPT Plus, Copilot Pro)? *

Markieren Sie nur ein Oval.

- ◯ Yes
- ◯ No, but I thought about it
- ◯ No, I don't see the reason for it currently
- ◯ No, I never will

10. How important are LLM chatbots in your daily workflow? *

Markieren Sie nur ein Oval.

- ◯ Essential
- ◯ Useful but not critical
- ◯ Nice to have but rarely needed
- ◯ Not important

Environmental Awareness

Now, we would like to explore your knowledge and perceptions of the energy use and environmental impact of conversational AI

11. On a scale of 1–5, how concerned are you about the environmental impact of technology in general? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Not	◯	◯	◯	◯	◯	Extremely concerned

12. Are you aware that LLM chatbots consume significant energy and have a massive environmental impact? *

Markieren Sie nur ein Oval.

- ◯ Yes, I am well aware
- ◯ I am somewhat aware
- ◯ No, I did not know

13. Compared to a typical Google search, how much energy do you believe a single ChatGPT prompt consumes roughly? *

Markieren Sie nur ein Oval.

- ◯ Less energy than a typical Google search
- ◯ About the same amount of energy as a Google search
- ◯ Around 2× more energy
- ◯ Around 5× more energy
- ◯ Around 10× more energy
- ◯ I'm not sure / I can't estimate

14. How many ChatGPT queries do you think use roughly the same amount of energy as to fully charge an iPhone 14 (from 0% to 100%)? *

Markieren Sie nur ein Oval.

- ◯ About 20 queries
- ◯ About 50 queries
- ◯ About 100 queries
- ◯ About 1000 queries
- ◯ About 10'000 queries

15. Do you agree that LLM chatbots should generally be optimised to reduce energy consumption? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Stro	◯	◯	◯	◯	◯	Strongly agree

16. Currently, AI companies don't disclose a lot of information about the energy consumption of their models. Do you agree that AI companies should be more transparent about the environmental impact of their models and products? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Stro	◯	◯	◯	◯	◯	Strongly agree

Reimagine Conversational AI/chatbots

Finally, we'd love your input on potential sustainability features and how they might influence your usage.

17. On a scale of 1–5, how important is the environmental impact of conversational AI in your decision to use any such services? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Not	◯	◯	◯	◯	◯	Extremely important

18. Would you like LLM chatbots to provide an "Eco Mode" that reduces computational power for less demanding queries? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Not	◯	◯	◯	◯	◯	Extremely interested

19. Would you prefer to use an LLM chatbot that demonstrates a smaller carbon footprint, even if it is slower or less feature-rich? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Do r	◯	◯	◯	◯	◯	Strongly prefer

20. How important is it for you to see energy consumption information related to your conversational AI usage? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Not	◯	◯	◯	◯	◯	Extremely important

21. If such usage information was provided, would it influence how you use LLM chatbots? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
It wc	◯	◯	◯	◯	◯	It would strongly influence

22. Would you like to set limits on your LLM chatbot usage based on environmental impact? *

Markieren Sie nur ein Oval.

- ◯ Yes
- ◯ No
- ◯ Unsure

23. Would you agree to pay a small fee or donation to offset the carbon emissions of your conversational AI usage? *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
Stro	◯	◯	◯	◯	◯	Strongly agree

24. Do you have any other thoughts on how conversational AI could be optimized for sustainability?

About you

Before wrapping everything up, please tell us a bit about yourself.

25. Which age group do you belong to? *

Markieren Sie nur ein Oval.

- ◯ 18 - 29
- ◯ 30 - 44
- ◯ 45 - 60
- ◯ above 60

26. What is your gender? *

Markieren Sie nur ein Oval.

- ◯ Male
- ◯ Female
- ◯ Diverse
- ◯ Sonstiges: _____

27. Which of the following best describes your current role in your organization? *

Markieren Sie nur ein Oval.

- ◯ Student / Intern
- ◯ Academic / Researcher / Educator
- ◯ Developer / Engineer
- ◯ Data Scientist / Analyst
- ◯ Manager / Team Lead
- ◯ Consultant / Advisor
- ◯ Operations / Administrative
- ◯ Non-technical / Business-focused
- ◯ Sonstiges: _____

28. In which business domain are you primarily working currently? *

Markieren Sie nur ein Oval.

- ◯ Education / Academia
- ◯ Software / IT Services
- ◯ AI / Machine Learning
- ◯ Finance / Banking
- ◯ Healthcare / Life Sciences
- ◯ Manufacturing / Industry
- ◯ Government / Public Sector
- ◯ Marketing / E-commerce
- ◯ Media / Entertainment
- ◯ Nonprofit / Social Sector
- ◯ Sonstiges: _____

29. ***That's it, thanks a ton!***

If you're interested in receiving the results of this survey please leave your email below.

_____

A.2 Experiment Forms and Results

A.2.1 Participant's Consent Document

Participant Consent Form for Participation of Experiment
Bachelor thesis
UI strategies for reducing conversational AI energy consumption

You are invited to participate in a study for our bachelor thesis: UI strategies for reducing conversational AI energy consumption that is being conducted by Jack Gläser and Simon Lüscher, supervised by Prof. Martin Kropp and Dr. Nitish Patkar from University of Applied Sciences Northwestern Switzerland.

Contact email:
jack.glaeser@students.fhnw.ch
simon.luescher@students.fhnw.ch

1 Purpose
We are investigating whether specific user-interface features (energy note, 3-mode toggle, usage-metrics dashboard) raise users' awareness of the energy cost of large-language-model (LLM) prompts and nudge more sustainable usage patterns.

2 What participation involves

- **Duration:** 5 working days (Mon–Fri).
- **Use of prototype**: Chat with our web-based LLM assistant; all sustainability features are enabled by default.
- **Daily check-in**: One very short question micro-survey (< 1 min).
- **Final survey:** Single questionnaire (~5 min) at the end of Day 5.
  There are no interviews, screen recordings, audio or video captures

3 Data collected
Usage logs: prompt & response text, token counts, selected mode, feature toggles, timestamps. Encrypted on FHNW Azure servers; deleted 31 Dec 2025; anonymised stats kept 5 y
Surveys: daily check-ins, final questionnaire (awareness, usability, etc)

4 Voluntary participation
Taking part is entirely voluntary. You may skip questions or withdraw at any time without penalty; all identifiable data will then be erased.

5 Confidentiality
Data are stored under a coded participant-ID. Only the research team has access. Publications will report only aggregated, non-identifiable results.

6 Risks & benefits
The study poses minimal risk. You may gain insight into the energy footprint of everyday AI usage and help design more sustainable chatbots.

7 Results dissemination
Findings will appear in the bachelor thesis and may be submitted to academic venues. An anonymised dataset may be shared openly for reproducibility.

I have read and understood this information. I had the opportunity to ask questions and received satisfactory answers. I voluntarily agree to:

1. Participate in this study.
2. Allow my anonymised data to be analysed and published as described.

Name and Signature **Date**

A.2.2 Daily check ins

IP6 - Experiment: Daily check-in

Daily check-in survey for participants of the Experiment.
Thanks a ton for filling it out! :-)

* Gibt eine erforderliche Frage an

1. E-Mail-Adresse *

2. At this moment I'm aware of the energy cost of the prompts I sent *today.* *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

3. The energy note shown under each response made me think about energy use. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

4. The energy estimate and 3-mode toggle influenced my choice of mode for today's prompts. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

5. I looked at the usage-metrics dashboard at least once today. *

Markieren Sie nur ein Oval.

- yes
- no

A.2.3 Final Questionnaire

IP6 - Experiment: Final Questionnaire

Hi there,

Thank you for completing the five-day study with our sustainable-AI chatbot. We have one final favour to ask: please fill out the **end-of-study questionnaire** linked below. It is a little longer than the daily check-ins (about 5–10 minutes) and will give us the insights we need to understand how the features worked for you.

We really appreciate the time and effort you've invested in this project.

Many thanks!

Jack & Simon

* Gibt eine erforderliche Frage an

1. E-Mail-Adresse *

2. I am aware that LLM AI-chatbots consume significant amount of energy *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

3. Compared with a Google search, a single ChatGPT-style prompt consumes more energy. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

4. It is important for me to know the energy cost of my AI-chatbot usage. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

Behaviour & feature experience

5. Knowing the rough energy costs of my prompts influences how I formulate prompts. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

6. I consciously shortened or reduced prompts after seeing energy-related information. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

7. I actively choose Eco-mode when high accuracy was not required. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

8. I checked the sustainability dashboard(Usage metrics) at least 2 times within the period. *

Markieren Sie nur ein Oval.

- yes
- no
- not sure

9. The energy-related information distracted me from completing my main task. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

10. I would like to see similar sustainability features in our mainstream AI-chatbots. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

11. Which of the experiences features within the experiment would you like to see adopted by mainstream AI-Chatbots?

Wählen Sie alle zutreffenden Antworten aus.

- [ ] Energy info-note on the responses
- [ ] Different models / being able to switch models fast based on input
- [ ] Usage metrics / some sort of dashboard with usage /energy consumption data
- [ ] Estimation on energy costs while typing out a prompt
- [ ] None of them

Usability & satisfaction

12. The energy note (refering to how many X-units it took) was easy to understand and its effect was clear. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

13. The 3-mode toggle was easy to understand and its effect was clear. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

14. The usage metrics dashboard was easy to understand and its effect was clear *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

15. The usage metrics dashboard was easy to understand *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

16. The usage metrics dashboard helped to increase my overall knowledge on energy consumption of AI-Chatbot usage *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

17. The placement of the “choose your AI mode” was well placed and easy to understand. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

18. The features overall fit well with how I usually work with AI-chatbots and did not further distract me a lot from using the AI-chatbot. *

Markieren Sie nur ein Oval.

	1	2	3	4	5	
stro	◯	◯	◯	◯	◯	strongly agree

Open questions

19. Do you have suggestion on raising energy usage or making energy usage more actionable in future AI-chatbot tools? / Do you have a feature idea? 🙂 *

20. Roughly how many AI-chatbot prompts per day do you make *outside* this prototype? *(numeric)* *

A.3 JSON Schema

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "User",
  "type": "object",
  "description": "Represents an application user with sustainability
     settings and preferences.",
  "properties": {
    "id": {
      "type": "string",
      "description": "Unique identifier of the user entity."
    },
    "userId": {
      "type": "string",
      "description": "Partition key: ID linking this entity to the user
         account."
    },
    "createdAt": {
      "type": "string",
      "format": "date-time",
      "description": "Timestamp when the user record was created."
    },
    "name": {
      "type": "string",
      "description": "Display name of the user."
    },
    "identity": {
      "type": "string",
      "minLength": 1,
      "description": "External or internal identity identifier."
    },
    "enableSustainabilityFeatures": {
      "type": "boolean",
      "description": "Indicates whether sustainability features are enabled
         ."
    },
    "mode": {
      "type": "number",
      "enum": [ 0, 1, 2 ],
      "description": "Chat mode preference for the user. 0=EnergyEfficient,
          1=Balanced, 2=Performance"
    },
    "energyUnitId": {
      "type": [ "string", "null" ],
      "description": "Reference ID to the user's selected energy unit."
    }
  },
  "required": [
    "id",
    "userId",
    "createdAt",
    "name",
    "identity",
    "enableSustainabilityFeatures",
    "mode"
  ],
```

```
  "additionalProperties": false
}
```

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Conversation",
  "type": "object",
  "description": "Represents a user's conversation with the chat bot.",
  "properties": {
    "id": {
      "type": "string",
      "description": "Unique identifier of the conversation."
    },
    "userId": {
      "type": "string",
      "description": "Partition key: User ID that owns the conversation."
    },
    "createdAt": {
      "type": "string",
      "format": "date-time",
      "description": "Timestamp when the conversation was created."
    },
    "title": {
      "type": "string",
      "description": "Title for the conversation."
    },
    "lastMessageTimestamp": {
      "type": [ "string", "null" ],
      "format": "date-time",
      "description": "Timestamp of the most recent message."
    },
    "isDeleted": {
      "type": [ "boolean", "null" ],
      "description": "Whether the conversation is marked as deleted."
    }
  },
  "required": [ "id", "userId", "createdAt", "title" ],
  "additionalProperties": false
}
```

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Prompt",
  "type": "object",
  "description": "Represents a user prompt and its associated metadata.",
  "properties": {
    "id": {
      "type": "string",
      "description": "Unique identifier of the prompt entity."
    },
    "userId": {
      "type": "string",
      "description": "Partition key: ID of the user who submitted the
         prompt."
    },
    "createdAt": {
      "type": "string",
```

```
      "format": "date-time",
      "description": "Timestamp when the prompt was created."
    },
    "conversationId": {
      "type": [ "string", "null" ],
      "description": "ID of the conversation this prompt belongs to."
    },
    "promptTextHistory": {
      "type": [ "array", "null" ],
      "items": {
        "type": "string"
      },
      "description": "List of previous (not sent) texts of this prompt."
    },
    "userText": {
      "type": [ "string", "null" ],
      "description": "Prompt text provided by the user."
    },
    "chatMode": {
      "type": "number",
      "enum": [ 0, 1, 2 ],
      "description": "Operational mode used for this prompt. 0=
          EnergyEfficient, 1=Balanced, 2=Performance"
    },
    "modelName": {
      "type": "string",
      "description": "Name of the model used to generate the response."
    },
    "historyLimit": {
      "type": "integer",
      "description": "Maximum number of previous prompts included in the
          request."
    },
    "isSent": {
      "type": [ "boolean", "null" ],
      "description": "Indicates whether the prompt was successfully sent."
    },
    "sentAt": {
      "type": [ "string", "null" ],
      "format": "date-time",
      "description": "Timestamp when the prompt was sent."
    },
    "responseText": {
      "type": [ "string", "null" ],
      "description": "Text response generated for the prompt."
    },
    "usage": {
      "type": [ "object", "null" ],
      "description": "Represents token usage and associated energy
          consumption.",
      "properties": {
        "numberOfInputTokens": {
          "type": "integer",
          "description": "Number of input tokens provided by the user."
        },
        "numberOfOutputTokens": {
          "type": "integer",
```

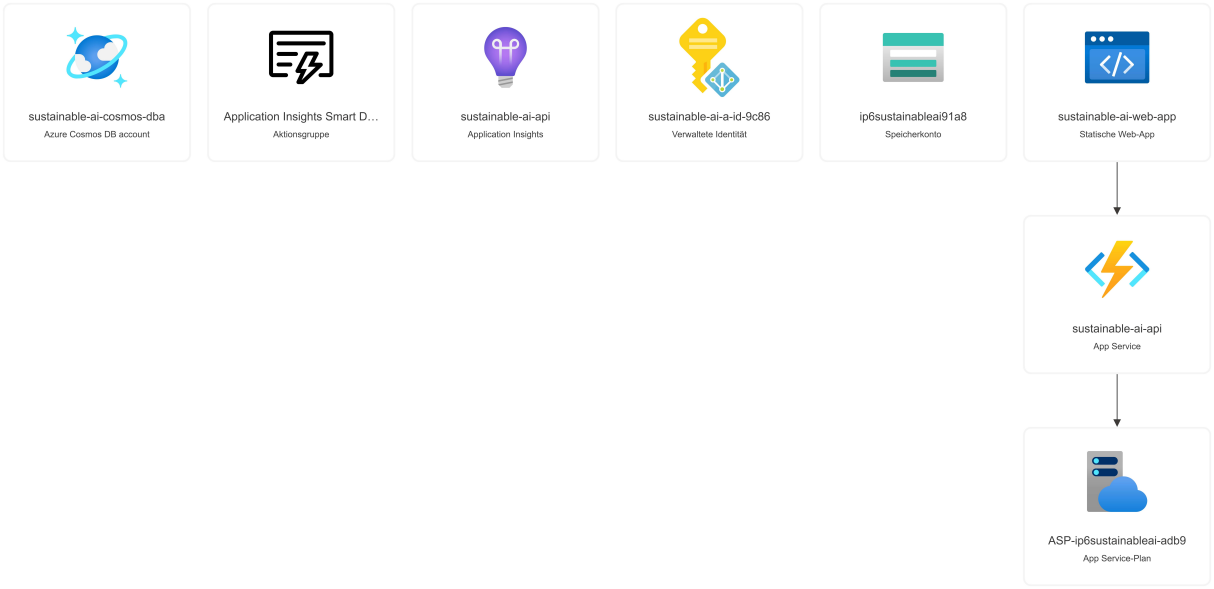
```
          "description": "Number of output tokens generated in response."
        },
        "usageInWh": {
          "type": "number",
          "description": "Estimated energy usage in watt-hours (Wh)."
        }
      },
      "required": [ "numberOfInputTokens", "numberOfOutputTokens", "
         usageInWh" ],
      "additionalProperties": false
    }
  },
  "required": [ "id", "userId", "createdAt", "chatMode", "modelName", "
     historyLimit" ],
  "additionalProperties": false
}
```

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Log",
  "type": "object",
  "description": "Represents a log entry.",
  "properties": {
    "id": {
      "type": "string",
      "description": "Unique identifier of the log entry."
    },
    "userId": {
      "type": "string",
      "description": "Partition key: User ID associated with the log."
    },
    "createdAt": {
      "type": "string",
      "format": "date-time",
      "description": "Timestamp when the log was created."
    },
    "type": {
      "type": "number",
      "enum": [ 0, 1, 2 ],
      "description": "Type/category of the log message. 0 = Unknown, 1 =
         PageVisit, 2 = SustainabilityModeChange."
    },
    "message": {
      "type": "string",
      "description": "Descriptive log message content."
    }
  },
  "required": [ "id", "userId", "createdAt", "type", "message" ],
  "additionalProperties": false
}
```

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "EnergyUnit",
  "type": "object",
  "description": "Defines a unit for measuring energy consumption.",
  "properties": {
```

```
    "id": {
      "type": "string",
      "description": "Unique identifier for the energy unit."
    },
    "tenantId": {
      "type": "string",
      "const": "0",
      "description": "Partition key: Fixed value \"0\""
    },
    "name": {
      "type": "string",
      "description": "Singular name of the energy unit (e.g., 'phone charge')."
    },
    "namePlural": {
      "type": "string",
      "description": "Plural form of the energy unit name (e.g., 'phone charges')."
    },
    "Wh": {
      "type": "number",
      "minimum": 0,
      "description": "Conversion factor to watt-hours (Wh)."
    }
  },
  "required": [ "id", "tenantId", "name", "namePlural", "Wh" ],
  "additionalProperties": false
}
```

A.3.1 Azure Infrastructure




```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/
       deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
```

```
        "sites_sustainable_ai_api_name": {
            "defaultValue": "sustainable-ai-api",
            "type": "String"
        },
        "components_sustainable_ai_api_name": {
            "defaultValue": "sustainable-ai-api",
            "type": "String"
        },
        "staticSites_sustainable_ai_web_app_name": {
            "defaultValue": "sustainable-ai-web-app",
            "type": "String"
        },
        "serverfarms_ASP_ip6sustainableai_adb9_name": {
            "defaultValue": "ASP-ip6sustainableai-adb9",
            "type": "String"
        },
        "storageAccounts_ip6sustainableai91a8_name": {
            "defaultValue": "ip6sustainableai91a8",
            "type": "String"
        },
        "databaseAccounts_sustainable_ai_cosmos_dba_name": {
            "defaultValue": "sustainable-ai-cosmos-dba",
            "type": "String"
        },
        "actionGroups_Application_Insights_Smart_Detection_name": {
            "defaultValue": "Application Insights Smart Detection",
            "type": "String"
        },
        "userAssignedIdentities_sustainable_ai_a_id_9c86_name": {
            "defaultValue": "sustainable-ai-a-id-9c86",
            "type": "String"
        },
        "workspaces_DefaultWorkspace_37baa613_2be4_4804_8a7e_c2c4a19538a5
           _CHN_externalid": {
            "defaultValue": "/subscriptions/37baa613-2be4-4804-8a7e-c2c4a19
               538a5/resourceGroups/DefaultResourceGroup-CHN/providers/
               Microsoft.OperationalInsights/workspaces/DefaultWorkspace-37
               baa613-2be4-4804-8a7e-c2c4a19538a5-CHN",
            "type": "String"
        }
    },
    "variables": {},
    "resources": [
        {
            "type": "Microsoft.DocumentDB/databaseAccounts",
            "apiVersion": "2024-12-01-preview",
            "name": "[parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name')]",
            "location": "Switzerland North",
            "tags": {
                "defaultExperience": "Core (SQL)",
                "hidden-workload-type": "Development/Testing",
                "hidden-cosmos-mmspecial": ""
            },
            "kind": "GlobalDocumentDB",
            "identity": {
                "type": "None"
```

```
            },
            "properties": {
                "publicNetworkAccess": "Enabled",
                "enableAutomaticFailover": true,
                "enableMultipleWriteLocations": false,
                "isVirtualNetworkFilterEnabled": false,
                "virtualNetworkRules": [],
                "disableKeyBasedMetadataWriteAccess": false,
                "enableFreeTier": false,
                "enableAnalyticalStorage": false,
                "analyticalStorageConfiguration": {
                    "schemaType": "WellDefined"
                },
                "databaseAccountOfferType": "Standard",
                "enableMaterializedViews": false,
                "capacityMode": "Serverless",
                "defaultIdentity": "FirstPartyIdentity",
                "networkAclBypass": "None",
                "disableLocalAuth": false,
                "enablePartitionMerge": false,
                "enablePerRegionPerPartitionAutoscale": false,
                "enableBurstCapacity": false,
                "enablePriorityBasedExecution": false,
                "defaultPriorityLevel": "High",
                "minimalTlsVersion": "Tls12",
                "consistencyPolicy": {
                    "defaultConsistencyLevel": "Session",
                    "maxIntervalInSeconds": 5,
                    "maxStalenessPrefix": 100
                },
                "locations": [
                    {
                        "locationName": "Switzerland North",
                        "failoverPriority": 0,
                        "isZoneRedundant": false
                    }
                ],
                "cors": [],
                "capabilities": [],
                "ipRules": [],
                "backupPolicy": {
                    "type": "Periodic",
                    "periodicModeProperties": {
                        "backupIntervalInMinutes": 240,
                        "backupRetentionIntervalInHours": 8,
                        "backupStorageRedundancy": "Geo"
                    }
                },
                "networkAclBypassResourceIds": [],
                "diagnosticLogSettings": {
                    "enableFullTextQuery": "None"
                },
                "capacity": {
                    "totalThroughputLimit": 4000
                }
            }
        },
```

```
        {
            "type": "microsoft.insights/actionGroups",
            "apiVersion": "2024-10-01-preview",
            "name": "[parameters('
               actionGroups_Application_Insights_Smart_Detection_name')]",
            "location": "Global",
            "properties": {
                "groupShortName": "SmartDetect",
                "enabled": true,
                "emailReceivers": [],
                "smsReceivers": [],
                "webhookReceivers": [],
                "eventHubReceivers": [],
                "itsmReceivers": [],
                "azureAppPushReceivers": [],
                "automationRunbookReceivers": [],
                "voiceReceivers": [],
                "logicAppReceivers": [],
                "azureFunctionReceivers": [],
                "armRoleReceivers": [
                    {
                        "name": "Monitoring Contributor",
                        "roleId": "749f88d5-cbae-40b8-bcfc-e573ddc772fa",
                        "useCommonAlertSchema": true
                    },
                    {
                        "name": "Monitoring Reader",
                        "roleId": "43d0d8ad-25c7-4714-9337-8ba259a9fe05",
                        "useCommonAlertSchema": true
                    }
                ]
            }
        },
        {
            "type": "microsoft.insights/components",
            "apiVersion": "2020-02-02",
            "name": "[parameters('components_sustainable_ai_api_name')]",
            "location": "switzerlandnorth",
            "kind": "web",
            "properties": {
                "Application_Type": "web",
                "Flow_Type": "Redfield",
                "Request_Source": "IbizaAIExtensionEnablementBlade",
                "RetentionInDays": 90,
                "WorkspaceResourceId": "[parameters('
                   workspaces_DefaultWorkspace_37baa613_2be4_4804_8a7e_c2c4
                   a19538a5_CHN_externalid')]",
                "IngestionMode": "LogAnalytics",
                "publicNetworkAccessForIngestion": "Enabled",
                "publicNetworkAccessForQuery": "Enabled"
            }
        },
        {
            "type": "Microsoft.ManagedIdentity/userAssignedIdentities",
            "apiVersion": "2025-01-31-preview",
            "name": "[parameters('
               userAssignedIdentities_sustainable_ai_a_id_9c86_name')]",
```

```
            "location": "switzerlandnorth"
        },
        {
            "type": "Microsoft.Storage/storageAccounts",
            "apiVersion": "2024-01-01",
            "name": "[parameters('storageAccounts_ip6sustainableai91a8_name')]",
            "location": "switzerlandnorth",
            "sku": {
                "name": "Standard_LRS",
                "tier": "Standard"
            },
            "kind": "Storage",
            "properties": {
                "defaultToOAuthAuthentication": true,
                "publicNetworkAccess": "Enabled",
                "allowCrossTenantReplication": false,
                "minimumTlsVersion": "TLS1_2",
                "allowBlobPublicAccess": false,
                "networkAcls": {
                    "bypass": "AzureServices",
                    "virtualNetworkRules": [],
                    "ipRules": [],
                    "defaultAction": "Allow"
                },
                "supportsHttpsTrafficOnly": true,
                "encryption": {
                    "services": {
                        "file": {
                            "keyType": "Account",
                            "enabled": true
                        },
                        "blob": {
                            "keyType": "Account",
                            "enabled": true
                        }
                    },
                    "keySource": "Microsoft.Storage"
                }
            }
        },
        {
            "type": "Microsoft.Web/serverfarms",
            "apiVersion": "2024-04-01",
            "name": "[parameters('serverfarms_ASP_ip6sustainableai_adb9_name')]",
            "location": "Switzerland North",
            "sku": {
                "name": "Y1",
                "tier": "Dynamic",
                "size": "Y1",
                "family": "Y",
                "capacity": 0
            },
            "kind": "functionapp",
            "properties": {
                "perSiteScaling": false,
```

```
                "elasticScaleEnabled": false,
                "maximumElasticWorkerCount": 1,
                "isSpot": false,
                "reserved": false,
                "isXenon": false,
                "hyperV": false,
                "targetWorkerCount": 0,
                "targetWorkerSizeId": 0,
                "zoneRedundant": false
            }
        },
        {
            "type": "Microsoft.Web/staticSites",
            "apiVersion": "2024-04-01",
            "name": "[parameters('staticSites_sustainable_ai_web_app_name')
               ]",
            "location": "West Europe",
            "sku": {
                "name": "Standard",
                "tier": "Standard"
            },
            "properties": {
                "repositoryUrl": "https://github.com/simonluescherfhnw/ip6-
                   sustainable-ai-frontend",
                "branch": "main",
                "stagingEnvironmentPolicy": "Enabled",
                "allowConfigFileUpdates": true,
                "provider": "GitHub",
                "enterpriseGradeCdnStatus": "Disabled"
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Development')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Development"
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Production')]",
            "dependsOn": [
```

```
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Production"
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/
                sqlRoleDefinitions",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/0000000
                0-0000-0000-0000-000000000001')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "roleName": "Cosmos DB Built-in Data Reader",
                "type": "BuiltInRole",
                "assignableScopes": [
                    "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                        parameters('
                        databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
                ],
                "permissions": [
                    {
                        "dataActions": [
                            "Microsoft.DocumentDB/databaseAccounts/
                                readMetadata",
                            "Microsoft.DocumentDB/databaseAccounts/
                                sqlDatabases/containers/executeQuery",
                            "Microsoft.DocumentDB/databaseAccounts/
                                sqlDatabases/containers/readChangeFeed",
                            "Microsoft.DocumentDB/databaseAccounts/
                                sqlDatabases/containers/items/read"
                        ],
                        "notDataActions": []
                    }
                ]
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/
                sqlRoleDefinitions",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/0000000
                0-0000-0000-0000-000000000002')]",
            "dependsOn": [
```

```
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "roleName": "Cosmos DB Built-in Data Contributor",
                "type": "BuiltInRole",
                "assignableScopes": [
                    "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                        parameters('
                        databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
                ],
                "permissions": [
                    {
                        "dataActions": [
                            "Microsoft.DocumentDB/databaseAccounts/
                                readMetadata",
                            "Microsoft.DocumentDB/databaseAccounts/
                                sqlDatabases/containers/*",
                            "Microsoft.DocumentDB/databaseAccounts/
                                sqlDatabases/containers/items/*"
                        ],
                        "notDataActions": []
                    }
                ]
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/
                tableRoleDefinitions",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/0000000
                0-0000-0000-0000-000000000001')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "roleName": "Cosmos DB Built-in Data Reader",
                "type": "BuiltInRole",
                "assignableScopes": [
                    "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                        parameters('
                        databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
                ],
                "permissions": [
                    {
                        "dataActions": [
                            "Microsoft.DocumentDB/databaseAccounts/
                                readMetadata",
                            "Microsoft.DocumentDB/databaseAccounts/tables/
                                containers/executeQuery",
                            "Microsoft.DocumentDB/databaseAccounts/tables/
                                containers/readChangeFeed",
```

```
                            "Microsoft.DocumentDB/databaseAccounts/tables/
                               containers/entities/read"
                        ],
                        "notDataActions": []
                    }
                ]
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/
               tableRoleDefinitions",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/0000000
               0-0000-0000-0000-000000000002')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "roleName": "Cosmos DB Built-in Data Contributor",
                "type": "BuiltInRole",
                "assignableScopes": [
                    "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                       parameters('
                       databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
                ],
                "permissions": [
                    {
                        "dataActions": [
                            "Microsoft.DocumentDB/databaseAccounts/
                               readMetadata",
                            "Microsoft.DocumentDB/databaseAccounts/tables/*
                               ",
                            "Microsoft.DocumentDB/databaseAccounts/tables/
                               containers/*",
                            "Microsoft.DocumentDB/databaseAccounts/tables/
                               containers/entities/*"
                        ],
                        "notDataActions": []
                    }
                ]
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/degradationindependencyduration')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
```

```
                "ruleDefinitions": {
                    "Name": "degradationindependencyduration",
                    "DisplayName": "Degradation in dependency duration",
                    "Description": "Smart Detection rules notify you of
                       performance anomaly issues.",
                    "HelpUrl": "https://docs.microsoft.com/en-us/azure/
                       application-insights/app-insights-proactive-
                       performance-diagnostics",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": false,
                    "SupportsEmailNotifications": true
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/degradationinserverresponsetime')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "degradationinserverresponsetime",
                    "DisplayName": "Degradation in server response time",
                    "Description": "Smart Detection rules notify you of
                       performance anomaly issues.",
                    "HelpUrl": "https://docs.microsoft.com/en-us/azure/
                       application-insights/app-insights-proactive-
                       performance-diagnostics",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": false,
                    "SupportsEmailNotifications": true
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/digestMailConfiguration')]",
            "location": "switzerlandnorth",
            "dependsOn": [
```

```
                "[resourceId('microsoft.insights/components', parameters('
                    components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "digestMailConfiguration",
                    "DisplayName": "Digest Mail Configuration",
                    "Description": "This rule describes the digest mail
                        preferences",
                    "HelpUrl": "www.homail.com",
                    "IsHidden": true,
                    "IsEnabledByDefault": true,
                    "IsInPreview": false,
                    "SupportsEmailNotifications": true
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
                ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
                '), '/extension_billingdatavolumedailyspikeextension')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                    components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_billingdatavolumedailyspikeextension
                        ",
                    "DisplayName": "Abnormal rise in daily data volume (
                        preview)",
                    "Description": "This detection rule automatically
                        analyzes the billing data generated by your
                        application, and can warn you about an unusual
                        increase in your application's billing costs",
                    "HelpUrl": "https://github.com/Microsoft/
                        ApplicationInsights-Home/tree/master/SmartDetection/
                        billing-data-volume-daily-spike.md",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
                ProactiveDetectionConfigs",
```

```
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name'), '/extension_canaryextension')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_canaryextension",
                    "DisplayName": "Canary extension",
                    "Description": "Canary extension",
                    "HelpUrl": "https://github.com/Microsoft/ApplicationInsights-Home/blob/master/SmartDetection/",
                    "IsHidden": true,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name'), '/extension_exceptionchangeextension')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_exceptionchangeextension",
                    "DisplayName": "Abnormal rise in exception volume (preview)",
                    "Description": "This detection rule automatically analyzes the exceptions thrown in your application, and can warn you about unusual patterns in your exception telemetry.",
                    "HelpUrl": "https://github.com/Microsoft/ApplicationInsights-Home/blob/master/SmartDetection/abnormal-rise-in-exception-volume.md",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
```

```
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/extension_memoryleakextension')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_memoryleakextension",
                    "DisplayName": "Potential memory leak detected (preview
                       )",
                    "Description": "This detection rule automatically
                       analyzes the memory consumption of each process in
                       your application, and can warn you about potential
                       memory leaks or increased memory consumption.",
                    "HelpUrl": "https://github.com/Microsoft/
                       ApplicationInsights-Home/tree/master/SmartDetection/
                       memory-leak.md",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/extension_securityextensionspackage')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_securityextensionspackage",
                    "DisplayName": "Potential security issue detected (
                       preview)",
                    "Description": "This detection rule automatically
                       analyzes the telemetry generated by your application
                        and detects potential security issues.",
                    "HelpUrl": "https://github.com/Microsoft/
                       ApplicationInsights-Home/blob/master/SmartDetection/
                       application-security-detection-pack.md",
```

```
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/extension_traceseveritydetector')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "extension_traceseveritydetector",
                    "DisplayName": "Degradation in trace severity ratio (
                       preview)",
                    "Description": "This detection rule automatically
                       analyzes the trace logs emitted from your
                       application, and can warn you about unusual patterns
                        in the severity of your trace telemetry.",
                    "HelpUrl": "https://github.com/Microsoft/
                       ApplicationInsights-Home/blob/master/SmartDetection/
                       degradation-in-trace-severity-ratio.md",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/
               ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name
               '), '/longdependencyduration')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('
                   components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "longdependencyduration",
```

```
                    "DisplayName": "Long dependency duration",
                    "Description": "Smart Detection rules notify you of performance anomaly issues.",
                    "HelpUrl": "https://docs.microsoft.com/en-us/azure/application-insights/app-insights-proactive-performance-diagnostics",
                    "IsHidden": false,
                    "IsEnabledByDefault": true,
                    "IsInPreview": false,
                    "SupportsEmailNotifications": true
                },
                "enabled": true,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name'), '/migrationToAlertRulesCompleted')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('components_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "ruleDefinitions": {
                    "Name": "migrationToAlertRulesCompleted",
                    "DisplayName": "Migration To Alert Rules Completed",
                    "Description": "A configuration that controls the migration state of Smart Detection to Smart Alerts",
                    "HelpUrl": "https://docs.microsoft.com/en-us/azure/application-insights/app-insights-proactive-performance-diagnostics",
                    "IsHidden": true,
                    "IsEnabledByDefault": false,
                    "IsInPreview": true,
                    "SupportsEmailNotifications": false
                },
                "enabled": false,
                "sendEmailsToSubscriptionOwners": true,
                "customEmails": []
            }
        },
        {
            "type": "microsoft.insights/components/ProactiveDetectionConfigs",
            "apiVersion": "2018-05-01-preview",
            "name": "[concat(parameters('components_sustainable_ai_api_name'), '/slowpageloadtime')]",
            "location": "switzerlandnorth",
            "dependsOn": [
                "[resourceId('microsoft.insights/components', parameters('components_sustainable_ai_api_name'))]"
            ],
```

```
652             "properties": {
653                 "ruleDefinitions": {
654                     "Name": "slowpageloadtime",
655                     "DisplayName": "Slow page load time",
656                     "Description": "Smart Detection rules notify you of
                        performance anomaly issues.",
657                     "HelpUrl": "https://docs.microsoft.com/en-us/azure/
                        application-insights/app-insights-proactive-
                        performance-diagnostics",
658                     "IsHidden": false,
659                     "IsEnabledByDefault": true,
660                     "IsInPreview": false,
661                     "SupportsEmailNotifications": true
662                 },
663                 "enabled": true,
664                 "sendEmailsToSubscriptionOwners": true,
665                 "customEmails": []
666             }
667         },
668         {
669             "type": "microsoft.insights/components/
                ProactiveDetectionConfigs",
670             "apiVersion": "2018-05-01-preview",
671             "name": "[concat(parameters('components_sustainable_ai_api_name
                '), '/slowserverresponsetime')]",
672             "location": "switzerlandnorth",
673             "dependsOn": [
674                 "[resourceId('microsoft.insights/components', parameters('
                    components_sustainable_ai_api_name'))]"
675             ],
676             "properties": {
677                 "ruleDefinitions": {
678                     "Name": "slowserverresponsetime",
679                     "DisplayName": "Slow server response time",
680                     "Description": "Smart Detection rules notify you of
                        performance anomaly issues.",
681                     "HelpUrl": "https://docs.microsoft.com/en-us/azure/
                        application-insights/app-insights-proactive-
                        performance-diagnostics",
682                     "IsHidden": false,
683                     "IsEnabledByDefault": true,
684                     "IsInPreview": false,
685                     "SupportsEmailNotifications": true
686                 },
687                 "enabled": true,
688                 "sendEmailsToSubscriptionOwners": true,
689                 "customEmails": []
690             }
691         },
692         {
693             "type": "Microsoft.ManagedIdentity/userAssignedIdentities/
                federatedIdentityCredentials",
694             "apiVersion": "2025-01-31-preview",
695             "name": "[concat(parameters('
                userAssignedIdentities_sustainable_ai_a_id_9c86_name'), '/
                simonluescherfhnw-ip6-sustainable-ai-backend-a416')]",
696             "dependsOn": [
```

```
                "[resourceId('Microsoft.ManagedIdentity/
                    userAssignedIdentities', parameters('
                    userAssignedIdentities_sustainable_ai_a_id_9c86_name'))]
                    "
            ],
            "properties": {
                "issuer": "https://token.actions.githubusercontent.com",
                "subject": "repo:simonluescherfhnw/ip6-sustainable-ai-
                    backend:ref:refs/heads/main",
                "audiences": [
                    "api://AzureADTokenExchange"
                ]
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/blobServices",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "sku": {
                "name": "Standard_LRS",
                "tier": "Standard"
            },
            "properties": {
                "cors": {
                    "corsRules": []
                },
                "deleteRetentionPolicy": {
                    "allowPermanentDelete": false,
                    "enabled": false
                }
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/fileServices",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "sku": {
                "name": "Standard_LRS",
                "tier": "Standard"
            },
            "properties": {
                "protocolSettings": {
                    "smb": {}
                },
                "cors": {
```

```
                    "corsRules": []
                },
                "shareDeleteRetentionPolicy": {
                    "enabled": true,
                    "days": 7
                }
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/queueServices",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
               a8_name'), '/default')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts',
                   parameters('storageAccounts_ip6sustainableai91a8_name'))
                   ]"
            ],
            "properties": {
                "cors": {
                    "corsRules": []
                }
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/tableServices",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
               a8_name'), '/default')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts',
                   parameters('storageAccounts_ip6sustainableai91a8_name'))
                   ]"
            ],
            "properties": {
                "cors": {
                    "corsRules": []
                }
            }
        },
        {
            "type": "Microsoft.Web/sites",
            "apiVersion": "2024-04-01",
            "name": "[parameters('sites_sustainable_ai_api_name')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/serverfarms', parameters('
                   serverfarms_ASP_ip6sustainableai_adb9_name'))]"
            ],
            "kind": "functionapp",
            "properties": {
                "enabled": true,
                "hostNameSslStates": [
                    {
                        "name": "[concat(parameters('
                           sites_sustainable_ai_api_name'), '.azurewebsites
                           .net')]",
```

```
                        "sslState": "Disabled",
                        "hostType": "Standard"
                    },
                    {
                        "name": "[concat(parameters('
                           sites_sustainable_ai_api_name'), '.scm.
                           azurewebsites.net')]",
                        "sslState": "Disabled",
                        "hostType": "Repository"
                    }
                ],
                "serverFarmId": "[resourceId('Microsoft.Web/serverfarms',
                   parameters('serverfarms_ASP_ip6sustainableai_adb9_name')
                   )]",
                "reserved": false,
                "isXenon": false,
                "hyperV": false,
                "dnsConfiguration": {},
                "vnetRouteAllEnabled": false,
                "vnetImagePullEnabled": false,
                "vnetContentShareEnabled": false,
                "siteConfig": {
                    "numberOfWorkers": 1,
                    "acrUseManagedIdentityCreds": false,
                    "alwaysOn": false,
                    "http20Enabled": false,
                    "functionAppScaleLimit": 200,
                    "minimumElasticInstanceCount": 0
                },
                "scmSiteAlsoStopped": false,
                "clientAffinityEnabled": false,
                "clientCertEnabled": false,
                "clientCertMode": "Required",
                "hostNamesDisabled": false,
                "ipMode": "IPv4",
                "vnetBackupRestoreEnabled": false,
                "customDomainVerificationId": "12AB493E38ABD001261CF21640AB
                   413738832F9779A1C425D06A0380761D8BA9",
                "containerSize": 1536,
                "dailyMemoryTimeQuota": 0,
                "httpsOnly": true,
                "endToEndEncryptionEnabled": false,
                "redundancyMode": "None",
                "publicNetworkAccess": "Enabled",
                "storageAccountRequired": false,
                "keyVaultReferenceIdentity": "SystemAssigned"
            }
        },
        {
            "type": "Microsoft.Web/sites/basicPublishingCredentialsPolicies
               ",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/ftp')]",
            "location": "Switzerland North",
            "dependsOn": [
```

```
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "allow": false
            }
        },
        {
            "type": "Microsoft.Web/sites/basicPublishingCredentialsPolicies
                ",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/scm')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "allow": false
            }
        },
        {
            "type": "Microsoft.Web/sites/config",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/web')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "numberOfWorkers": 1,
                "defaultDocuments": [
                    "Default.htm",
                    "Default.html",
                    "Default.asp",
                    "index.htm",
                    "index.html",
                    "iisstart.htm",
                    "default.aspx",
                    "index.php"
                ],
                "netFrameworkVersion": "v8.0",
                "requestTracingEnabled": false,
                "remoteDebuggingEnabled": false,
                "httpLoggingEnabled": false,
                "acrUseManagedIdentityCreds": false,
                "logsDirectorySizeLimit": 35,
                "detailedErrorLoggingEnabled": false,
                "publishingUsername": "REDACTED",
                "scmType": "GitHubAction",
                "use32BitWorkerProcess": false,
                "webSocketsEnabled": false,
                "alwaysOn": false,
                "managedPipelineMode": "Integrated",
```

```
                "virtualApplications": [
                    {
                        "virtualPath": "/",
                        "physicalPath": "site\\wwwroot",
                        "preloadEnabled": false
                    }
                ],
                "loadBalancing": "LeastRequests",
                "experiments": {
                    "rampUpRules": []
                },
                "autoHealEnabled": false,
                "vnetRouteAllEnabled": false,
                "vnetPrivatePortsCount": 0,
                "publicNetworkAccess": "Enabled",
                "cors": {
                    "allowedOrigins": [
                        "https://portal.azure.com",
                        "https://green-mud-04afae203.6.azurestaticapps.net"
                    ],
                    "supportCredentials": false
                },
                "localMySqlEnabled": false,
                "ipSecurityRestrictions": [
                    {
                        "ipAddress": "Any",
                        "action": "Allow",
                        "priority": 2147483647,
                        "name": "Allow all",
                        "description": "Allow all access"
                    }
                ],
                "scmIpSecurityRestrictions": [
                    {
                        "ipAddress": "Any",
                        "action": "Allow",
                        "priority": 2147483647,
                        "name": "Allow all",
                        "description": "Allow all access"
                    }
                ],
                "scmIpSecurityRestrictionsUseMain": false,
                "http20Enabled": false,
                "minTlsVersion": "1.2",
                "scmMinTlsVersion": "1.2",
                "ftpsState": "FtpsOnly",
                "preWarmedInstanceCount": 0,
                "functionAppScaleLimit": 200,
                "functionsRuntimeScaleMonitoringEnabled": false,
                "minimumElasticInstanceCount": 0,
                "azureStorageAccounts": {}
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
```

```
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/076c043f429d4b8ab61a0a615df58b07')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"419f7ebecb79
                    48f46bf9078a0528fc7514af3f81\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-23T15:01:11.4681421Z",
                "end_time": "2025-06-23T15:01:12.8275307Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/2338922e33cf4e73bec6c487c2719e5b')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"4ca2c2e9bc10
                    1412c924390186decd0ee39e4a08\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T20:06:26.235006Z",
                "end_time": "2025-06-22T20:06:27.7987243Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/23bb4f51bfa244c9b15bb32eee71b5f1')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
```

```
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"1b590ce92de3
                    11c631be7d64d636f067b64a405f\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T21:22:34.294617Z",
                "end_time": "2025-06-22T21:22:35.5289937Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/435e372ef56847438fe34fe62aafeb84')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"e6c8c842e11a
                    4e1ffdf5f04518dc099f8489d676\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T18:13:00.8034236Z",
                "end_time": "2025-06-22T18:13:02.335107Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/60c0227b3c3b45b7bcdffb5c6b0c2e3e')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"2532b70ad018
                    ca5fbf8c8da5e015a965cd3ed196\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T15:06:30.2846682Z",
                "end_time": "2025-06-22T15:06:31.7455107Z",
```

```
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/731c60269c7d4c478be50128397d5f3a')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                   sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"c15f9163235f
                   02835ad786a184571713476ee7ad\",\"repoName\":\"
                   simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                   :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T13:05:18.8184065Z",
                "end_time": "2025-06-22T13:05:20.4590363Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/77373ccd6d6b41468229de0efcc5f418')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                   sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"9fee899ff89c
                   096f832a0fe66c993dd6df09b391\",\"repoName\":\"
                   simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                   :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T15:36:23.1319342Z",
                "end_time": "2025-06-22T15:36:24.7725459Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/c46f976169fd44ec9fcd4498c602cffc')]",
            "location": "Switzerland North",
```

```
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"3e471ef8240
                    eb2b06363ac53bd75717beb281f59\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T20:53:56.0463655Z",
                "end_time": "2025-06-22T20:53:57.5153299Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/d7386b0db21443178515e424abbffda3')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
                "message": "{\"type\":\"deployment\",\"sha\":\"3766cbf208f6
                    95f7ade3216e16fc2363650bc154\",\"repoName\":\"
                    simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\"
                    :\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-22T18:22:19.429819Z",
                "end_time": "2025-06-22T18:22:21.682451Z",
                "active": false
            }
        },
        {
            "type": "Microsoft.Web/sites/deployments",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/f978abb7e53b40b9bfba2cb195bb857e')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "status": 4,
                "author_email": "N/A",
                "author": "N/A",
                "deployer": "GITHUB_ZIP_DEPLOY_FUNCTIONS_V1",
```

```
                "message": "{\"type\":\"deployment\",\"sha\":\"9a8019effc5d17f37d528e0f2d3f75c6fd7d6942\",\"repoName\":\"simonluescherfhnw/ip6-sustainable-ai-backend\",\"actor\":\"simonluescherfhnw\",\"slotName\":\"production\"}",
                "start_time": "2025-06-30T17:19:09.0668586Z",
                "end_time": "2025-06-30T17:19:10.4049177Z",
                "active": true
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'), '/DeleteConversation')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites.net/admin/vfs/data/Functions/sampledata/DeleteConversation.dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin/functions/DeleteConversation",
                "config": {
                    "name": "DeleteConversation",
                    "entryPoint": "DeleteConversation.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.azurewebsites.net/api/deleteconversation",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
```

```
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/GetAppData')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                   sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                   net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                   .net/admin/vfs/data/Functions/sampledata/GetAppData.dat"
                   ,
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                   /functions/GetAppData",
                "config": {
                    "name": "GetAppData",
                    "entryPoint": "GetAppData.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "get"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                   azurewebsites.net/api/getappdata",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
               '/GetConversations')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                   sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
```

```
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/
                    GetConversations.dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/GetConversations",
                "config": {
                    "name": "GetConversations",
                    "entryPoint": "GetConversations.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/getconversations",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/GetPrompts')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/GetPrompts.dat"
                    ,
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/GetPrompts",
                "config": {
                    "name": "GetPrompts",
                    "entryPoint": "GetPrompts.Run",
```

```
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/getprompts",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/GetUsageStatistics')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/
                    GetUsageStatistics.dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/GetUsageStatistics",
                "config": {
                    "name": "GetUsageStatistics",
                    "entryPoint": "GetUsageStatistics.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
```

```
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/getusagestatistics",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/LogPageVisit')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/LogPageVisit.
                    dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/LogPageVisit",
                "config": {
                    "name": "LogPageVisit",
                    "entryPoint": "LogPageVisit.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
```

```
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/logpagevisit",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/PredictPromptUsage')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/
                    PredictPromptUsage.dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/PredictPromptUsage",
                "config": {
                    "name": "PredictPromptUsage",
                    "entryPoint": "PredictPromptUsage.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/predictpromptusage",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
```

```
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/SendPrompt')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/SendPrompt.dat"
                    ,
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/SendPrompt",
                "config": {
                    "name": "SendPrompt",
                    "entryPoint": "SendPrompt.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/sendprompt",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/UpdateConversation')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
```

```
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/
                    UpdateConversation.dat",
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/UpdateConversation",
                "config": {
                    "name": "UpdateConversation",
                    "entryPoint": "UpdateConversation.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/updateconversation",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/functions",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/UpdateUser')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "script_href": "https://sustainable-ai-api.azurewebsites.
                    net/admin/vfs/site/wwwroot/SustainableAI.Api.dll",
                "test_data_href": "https://sustainable-ai-api.azurewebsites
                    .net/admin/vfs/data/Functions/sampledata/UpdateUser.dat"
                    ,
                "href": "https://sustainable-ai-api.azurewebsites.net/admin
                    /functions/UpdateUser",
                "config": {
                    "name": "UpdateUser",
                    "entryPoint": "UpdateUser.Run",
                    "scriptFile": "SustainableAI.Api.dll",
                    "language": "dotnet-isolated",
```

```
                    "functionDirectory": "",
                    "bindings": [
                        {
                            "name": "req",
                            "type": "httpTrigger",
                            "direction": "In",
                            "authLevel": "Anonymous",
                            "methods": [
                                "post"
                            ]
                        },
                        {
                            "name": "$return",
                            "type": "http",
                            "direction": "Out"
                        }
                    ]
                },
                "invoke_url_template": "https://sustainable-ai-api.
                    azurewebsites.net/api/updateuser",
                "language": "dotnet-isolated",
                "isDisabled": false
            }
        },
        {
            "type": "Microsoft.Web/sites/hostNameBindings",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('sites_sustainable_ai_api_name'),
                '/', parameters('sites_sustainable_ai_api_name'), '.
                azurewebsites.net')]",
            "location": "Switzerland North",
            "dependsOn": [
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "siteName": "sustainable-ai-api",
                "hostNameType": "Verified"
            }
        },
        {
            "type": "Microsoft.Web/staticSites/basicAuth",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('
                staticSites_sustainable_ai_web_app_name'), '/default')]",
            "location": "West Europe",
            "dependsOn": [
                "[resourceId('Microsoft.Web/staticSites', parameters('
                    staticSites_sustainable_ai_web_app_name'))]"
            ],
            "properties": {
                "applicableEnvironmentsMode": "SpecifiedEnvironments"
            }
        },
        {
            "type": "Microsoft.Web/staticSites/customDomains",
            "apiVersion": "2024-04-01",
```

```
            "name": "[concat(parameters('
               staticSites_sustainable_ai_web_app_name'), '/thebotter.com')
               ]",
            "location": "West Europe",
            "dependsOn": [
                "[resourceId('Microsoft.Web/staticSites', parameters('
                   staticSites_sustainable_ai_web_app_name'))]"
            ],
            "properties": {}
        },
        {
            "type": "Microsoft.Web/staticSites/customDomains",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('
               staticSites_sustainable_ai_web_app_name'), '/www.thebotter.
               com')]",
            "location": "West Europe",
            "dependsOn": [
                "[resourceId('Microsoft.Web/staticSites', parameters('
                   staticSites_sustainable_ai_web_app_name'))]"
            ],
            "properties": {}
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Development/Conversation')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Development')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Conversation",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
```

```
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Production/Conversation')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Production')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Conversation",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
```

```
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Development/EnergyUnit')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Development')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "EnergyUnit",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/tenantId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
```

```
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Production/EnergyUnit')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Production')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "EnergyUnit",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/tenantId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
```

```
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
                containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
                Development/Log')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                    sqlDatabases', parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                    Development')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Log",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
                containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
                Production/Log')]",
```

```
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                    sqlDatabases', parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                    Production')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Log",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
                containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
                Development/Prompt')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                    sqlDatabases', parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                    Development')]",
```

```
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "Prompt",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
                containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
                databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
                Production/Prompt')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                    sqlDatabases', parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                    Production')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                    parameters('
                    databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
```

```
                    "id": "Prompt",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Development/User')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Development')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "User",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
```

```
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/
               containers",
            "apiVersion": "2024-12-01-preview",
            "name": "[concat(parameters('
               databaseAccounts_sustainable_ai_cosmos_dba_name'), '/
               Production/User')]",
            "dependsOn": [
                "[resourceId('Microsoft.DocumentDB/databaseAccounts/
                   sqlDatabases', parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'), '
                   Production')]",
                "[resourceId('Microsoft.DocumentDB/databaseAccounts',
                   parameters('
                   databaseAccounts_sustainable_ai_cosmos_dba_name'))]"
            ],
            "properties": {
                "resource": {
                    "id": "User",
                    "indexingPolicy": {
                        "indexingMode": "consistent",
                        "automatic": true,
                        "includedPaths": [
                            {
                                "path": "/*"
                            }
                        ],
                        "excludedPaths": [
                            {
                                "path": "/\"_etag\"/?"
```

```
                            }
                        ]
                    },
                    "partitionKey": {
                        "paths": [
                            "/userId"
                        ],
                        "kind": "Hash",
                        "version": 2
                    },
                    "uniqueKeyPolicy": {
                        "uniqueKeys": []
                    },
                    "conflictResolutionPolicy": {
                        "mode": "LastWriterWins",
                        "conflictResolutionPath": "/_ts"
                    },
                    "computedProperties": []
                }
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/blobServices/
               containers",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
               a8_name'), '/default/azure-webjobs-hosts')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts/
                   blobServices', parameters('storageAccounts_ip6
                   sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                   parameters('storageAccounts_ip6sustainableai91a8_name'))
                   ]"
            ],
            "properties": {
                "immutableStorageWithVersioning": {
                    "enabled": false
                },
                "defaultEncryptionScope": "$account-encryption-key",
                "denyEncryptionScopeOverride": false,
                "publicAccess": "None"
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/blobServices/
               containers",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
               a8_name'), '/default/azure-webjobs-secrets')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts/
                   blobServices', parameters('storageAccounts_ip6
                   sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                   parameters('storageAccounts_ip6sustainableai91a8_name'))
                   ]"
```

```
            ],
            "properties": {
                "immutableStorageWithVersioning": {
                    "enabled": false
                },
                "defaultEncryptionScope": "$account-encryption-key",
                "denyEncryptionScopeOverride": false,
                "publicAccess": "None"
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/fileServices/shares"
                ,
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default/sustainable-ai-api-35026538')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts/
                    fileServices', parameters('storageAccounts_ip6
                    sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "properties": {
                "accessTier": "TransactionOptimized",
                "shareQuota": 102400,
                "enabledProtocols": "SMB"
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/fileServices/shares"
                ,
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default/sustainable-ai-api-350265382f84')]",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts/
                    fileServices', parameters('storageAccounts_ip6
                    sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "properties": {
                "accessTier": "TransactionOptimized",
                "shareQuota": 102400,
                "enabledProtocols": "SMB"
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/fileServices/shares"
                ,
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default/sustainable-ai-apib12e')]",
            "dependsOn": [
```

```
                "[resourceId('Microsoft.Storage/storageAccounts/
                    fileServices', parameters('storageAccounts_ip6
                    sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "properties": {
                "accessTier": "TransactionOptimized",
                "shareQuota": 102400,
                "enabledProtocols": "SMB"
            }
        },
        {
            "type": "Microsoft.Storage/storageAccounts/tableServices/tables
                ",
            "apiVersion": "2024-01-01",
            "name": "[concat(parameters('storageAccounts_ip6sustainableai91
                a8_name'), '/default/AzureFunctionsDiagnosticEvents202507')]
                ",
            "dependsOn": [
                "[resourceId('Microsoft.Storage/storageAccounts/
                    tableServices', parameters('storageAccounts_ip6
                    sustainableai91a8_name'), 'default')]",
                "[resourceId('Microsoft.Storage/storageAccounts',
                    parameters('storageAccounts_ip6sustainableai91a8_name'))
                    ]"
            ],
            "properties": {}
        },
        {
            "type": "Microsoft.Web/staticSites/linkedBackends",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('
                staticSites_sustainable_ai_web_app_name'), '/backend1')]",
            "location": "West Europe",
            "dependsOn": [
                "[resourceId('Microsoft.Web/staticSites', parameters('
                    staticSites_sustainable_ai_web_app_name'))]",
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "backendResourceId": "[resourceId('Microsoft.Web/sites',
                    parameters('sites_sustainable_ai_api_name'))]",
                "region": "switzerlandnorth"
            }
        },
        {
            "type": "Microsoft.Web/staticSites/userProvidedFunctionApps",
            "apiVersion": "2024-04-01",
            "name": "[concat(parameters('
                staticSites_sustainable_ai_web_app_name'), '/backend1')]",
            "location": "West Europe",
            "dependsOn": [
                "[resourceId('Microsoft.Web/staticSites', parameters('
                    staticSites_sustainable_ai_web_app_name'))]",
```

```
                "[resourceId('Microsoft.Web/sites', parameters('
                    sites_sustainable_ai_api_name'))]"
            ],
            "properties": {
                "functionAppResourceId": "[resourceId('Microsoft.Web/sites'
                    , parameters('sites_sustainable_ai_api_name'))]",
                "functionAppRegion": "switzerlandnorth"
            }
        }
    ]
}
```

A.3.2 Matlab Scripts

```
%% === 0. Load files ===
file_prompts = 'Data_Pseudonym.xlsx';

% Read sheets into tables
prompts = readtable(file_prompts, 'Sheet', 'Prompts');
logs = readtable(file_prompts, 'Sheet', 'Logs');
conversations = readtable(file_prompts, 'Sheet', 'Conversations');
usersTable = readtable(file_prompts, 'Sheet', 'Users');

%% === 1. Preprocessing ===
% Interpret enums as categorical variables
prompts = prompts(prompts.isSent == true, :);
% Define custom chat mode order (including 'Total')
prompts.chatMode = categorical(prompts.chatMode, [0 1 2], {'Energy
    efficient', 'Balanced', 'Performance'});

% Convert timestamps to datetime
prompts.sentAt = datetime(prompts.sentAt, 'InputFormat', 'yyyy-MM-dd''T''HH
    :mm:ss');
prompts.createdAt = datetime(prompts.createdAt, 'InputFormat', 'yyyy-MM-dd
    ''T''HH:mm:ss');
%% === 2. Create new columns ===
prompts.responseLength = strlength(string(prompts.responseText));

%% === 3. Grouped evaluation (with all user-mode combos and totals) ===

% Get all unique users and all modes
allUsers = unique(prompts.userId);
allModes = categories(prompts.chatMode);

% Create full combination of users and modes
[U, M] = ndgrid(allUsers, allModes);
comboTable = table;
comboTable.userId = reshape(U, [], 1);
comboTable.chatMode = categorical(reshape(M, [], 1), allModes);

% Group actual data
G = findgroups(prompts.userId, prompts.chatMode);
T_actual = table;

T_actual.userId = splitapply(@(x) x(1), prompts.userId, G);
T_actual.chatMode = splitapply(@(x) x(1), prompts.chatMode, G);
```

```
T_actual.NumberOfPrompts = splitapply(@numel, prompts.id, G);
T_actual.InputTokens = splitapply(@sum, prompts.usage_numberOfInputTokens, G);
T_actual.OutputTokens = splitapply(@sum, prompts.usage_numberOfOutputTokens, G);
T_actual.TotalUsageWh = splitapply(@sum, prompts.usage_usageInWh, G);
T_actual.TotalUsageWhCorrected = splitapply(@sum, prompts.usageInWhCorrected, G);

% Join full combination with actual data to ensure 0s are included
T = outerjoin(comboTable, T_actual, ...
    'Keys', {'userId', 'chatMode'}, ...
    'MergeKeys', true);

% Replace NaNs with 0 for numeric columns
T.NumberOfPrompts(isnan(T.NumberOfPrompts)) = 0;
T.InputTokens(isnan(T.InputTokens)) = 0;
T.OutputTokens(isnan(T.OutputTokens)) = 0;
T.TotalUsageWh(isnan(T.TotalUsageWh)) = 0;
T.TotalUsageWhCorrected(isnan(T.TotalUsageWhCorrected)) = 0;

% Add total row per user
G_user = findgroups(T.userId);
T_userTotal = table;
T_userTotal.userId = splitapply(@(x) x(1), T.userId, G_user);
T_userTotal.chatMode = categorical(repmat("Total", height(T_userTotal), 1), ...
    [allModes; "Total"]);
T_userTotal.NumberOfPrompts = splitapply(@sum, T.NumberOfPrompts, G_user);
T_userTotal.InputTokens = splitapply(@sum, T.InputTokens, G_user);
T_userTotal.OutputTokens = splitapply(@sum, T.OutputTokens, G_user);
T_userTotal.TotalUsageWh = splitapply(@sum, T.TotalUsageWh, G_user);
T_userTotal.TotalUsageWhCorrected = splitapply(@sum, T.TotalUsageWhCorrected, G_user);

% Combine mode rows and total rows
T = [T; T_userTotal];

% Sort nicely by user, then mode
T = sortrows(T, {'userId', 'chatMode'});

% Add % column (only for modes, not total)
% Compute total prompts per user (one row per user)
userTotalPrompts = groupsummary(T, "userId", "max", "NumberOfPrompts");
userTotalPrompts.Properties.VariableNames{'max_NumberOfPrompts'} = 'TotalPromptsPerUser';

% Join this summary back into T
T = outerjoin(T, userTotalPrompts(:, {'userId', 'TotalPromptsPerUser'}), ...
    'Keys', 'userId', 'MergeKeys', true);
T.PctPromptsPerUserMode = 100 * (T.NumberOfPrompts ./ T.TotalPromptsPerUser);
T.TotalPromptsPerUser = [];  % Remove helper column from final table

T.PctPromptsPerUserMode(T.chatMode == "Total") = 100; % blank for totals

```

```
%% === 4. Display grouped results ===
disp(T);

%% === 5. Plot: Number of prompts per mode ===
figure('Name','Prompts per Mode');
modes = categories(T.chatMode);
counts = zeros(numel(modes),1);
for i = 1:numel(modes)
    counts(i) = sum(T.NumberOfPrompts(T.chatMode == modes{i}));
end
bar(categorical(modes), counts);
ylabel('Number of Prompts');
title('Total Prompts per Mode');
grid on;

%% === 6. Plot: Energy usage per user and mode ===
figure('Name','Energy Usage per User and Mode');
barData = unstack(T(:, {'userId', 'chatMode', 'TotalUsageWh'}), '
    TotalUsageWh', 'chatMode');
bar(categorical(barData.userId), barData{:,2:end});
xlabel('User ID');
ylabel('Energy Usage (Wh)');
title('Total Energy Usage per User & Mode');
legend(barData.Properties.VariableNames(2:end), 'Location', 'northwest');
grid on;

%% === 11. Display final table ===
disp('Evaluation per User and Mode:');
disp(T);

%% === 12. Aggregated evaluation per mode ===
[Gm, modes] = findgroups(T.chatMode);
Agg = table;

Agg.chatMode = modes;
Agg.InputTokens = splitapply(@sum, T.InputTokens, Gm);
Agg.OutputTokens = splitapply(@sum, T.OutputTokens, Gm);
Agg.TotalUsageWh = splitapply(@sum, T.TotalUsageWh, Gm);
Agg.TotalUsageWhCorrected = splitapply(@sum, T.TotalUsageWhCorrected, Gm);
Agg.TotalPrompts = splitapply(@sum, T.NumberOfPrompts, Gm);

% Calculate total prompts and total energy usage (use MAX because every
% mode also has a total)
totalPromptsAll = max(Agg.TotalPrompts);
totalWhAll = max(Agg.TotalUsageWh);
totalWhCorrectedAll = max(Agg.TotalUsageWhCorrected);

% Calculate percentage columns
Agg.PctPromptsPerMode = (Agg.TotalPrompts ./ totalPromptsAll) * 100;
Agg.PctUsagePerMode = (Agg.TotalUsageWh ./ totalWhAll) * 100;
Agg.PctUsageCorrectedPerMode = (Agg.TotalUsageWhCorrected ./
    totalWhCorrectedAll) * 100;

disp('Aggregated values per Mode:');
disp(Agg);

%% === Prompts per User and Day (Split by Mode and Total) ===
```

```
% Group by userId, day, and chatMode
G = findgroups(prompts.userId, prompts.day, prompts.chatMode);
summaryTable = table;
summaryTable.userId = splitapply(@(x) x(1), prompts.userId, G);
summaryTable.day = splitapply(@(x) x(1), prompts.day, G);
summaryTable.chatMode = splitapply(@(x) x(1), prompts.chatMode, G);
summaryTable.numPrompts = splitapply(@numel, prompts.id, G);

% Get all unique combinations
users = unique(prompts.userId);
days = unique(prompts.day);
modes = categories(prompts.chatMode);

% Generate variable names for each day-mode and day-total
varNames = {};
for d = days'
    for m = modes'
        varNames{end+1} = sprintf('Day%d_%s', d, matlab.lang.makeValidName(
            char(m)));
    end
    varNames{end+1} = sprintf('Day%d_Total', d);
end

% Initialize result table
result = array2table(zeros(numel(users), numel(varNames)), ...
    'VariableNames', varNames);
result.userId = users;

% Fill in prompt counts per user/day/mode
for i = 1:height(summaryTable)
    uid = summaryTable.userId(i);
    day = summaryTable.day(i);
    mode = summaryTable.chatMode(i);
    n = summaryTable.numPrompts(i);

    rowIdx = find(result.userId == uid);
    colName = sprintf('Day%d_%s', day, matlab.lang.makeValidName(char(mode)
        ));
    result{rowIdx, colName} = result{rowIdx, colName} + n;

    % Update total
    totalCol = sprintf('Day%d_Total', day);
    result{rowIdx, totalCol} = result{rowIdx, totalCol} + n;
end

% Move userId to the front
result = movevars(result, 'userId', 'Before', 1);

% Display result
disp('Prompt Matrix per User, Day, and Mode with Totals:');
disp(result);

%% === Prompts per User and Day ===
% Group prompts by userId and day, count number of prompts
G_day = findgroups(prompts.userId, prompts.day);
userDayTable = table;

```

```
userDayTable.userId = splitapply(@(x) x(1), prompts.userId, G_day);
userDayTable.day = splitapply(@(x) x(1), prompts.day, G_day);
userDayTable.NumberOfPrompts = splitapply(@numel, prompts.id, G_day);

% Display the table
disp('Number of prompts per User and Day:');
disp(userDayTable);

% Unique users and days
users = unique(userDayTable.userId);
days = unique(userDayTable.day);

% Build prompt matrix: rows = users, columns = days
promptMatrix = zeros(numel(users), numel(days));
for i = 1:numel(users)
    for j = 1:numel(days)
        idx = userDayTable.userId == users(i) & userDayTable.day == days(j)
            ;
        if any(idx)
            promptMatrix(i,j) = userDayTable.NumberOfPrompts(idx);
        end
    end
end

% === Grouped Bar Chart: Per User (X) and Day (grouped bars) ===
figure('Name','Prompts per User and Day (Bar Chart)');
bar(users, promptMatrix, 'grouped');
xlabel('User ID');
ylabel('Number of Prompts');
title('Number of Prompts per User and Day');
legend(arrayfun(@(d) sprintf('Day %d', d), days, 'UniformOutput', false),
    ...
        'Location', 'northeastoutside');
grid on;

% === Line Chart: Per User (X) and Day (one line per day) ===
figure('Name', 'Prompts per User and Day (Line Chart)');
hold on;

for j = 1:numel(days)
    plot(users, promptMatrix(:,j), '-o', 'DisplayName', sprintf('Day %d',
        days(j)));
end

% Plot average across days
avgPrompts = mean(promptMatrix, 2);
plot(users, avgPrompts, '-k', 'LineWidth', 2, 'DisplayName', 'Average');

xlabel('User ID');
ylabel('Number of Prompts');
title('Number of Prompts per User and Day with Average');
legend('Location', 'northeastoutside');
grid on;
hold off;
```

```
%% === Plot: Chat Mode Usage per Day ===

% Group by day and chat mode
G_dayMode = findgroups(prompts.day, prompts.chatMode);
modeDayTable = table;
modeDayTable.day = splitapply(@(x) x(1), prompts.day, G_dayMode);
modeDayTable.chatMode = splitapply(@(x) x(1), prompts.chatMode, G_dayMode);
modeDayTable.NumberOfPrompts = splitapply(@numel, prompts.id, G_dayMode);

% Prepare matrix: rows = days, columns = modes
days = unique(modeDayTable.day);
modes = categories(prompts.chatMode);
modeMatrix = zeros(numel(days), numel(modes));

for i = 1:numel(days)
    for j = 1:numel(modes)
        idx = (modeDayTable.day == days(i)) & (modeDayTable.chatMode ==
           modes{j});
        if any(idx)
            modeMatrix(i, j) = modeDayTable.NumberOfPrompts(idx);
        else
            modeMatrix(i, j) = 0;
        end
    end
end

% Plot stacked bar chart
figure('Name','Chat Mode Usage per Day');
bar(days, modeMatrix, 'stacked');
xlabel('Day');
ylabel('Number of Prompts');
title('Chat Mode Usage per Day');
legend(modes, 'Location', 'northeastoutside');
grid on;

%% === Plot: Chat Mode Usage per Day (Percentage) ===

% Normalize modeMatrix to percentages
modeMatrixPct = modeMatrix ./ sum(modeMatrix, 2) * 100;

% Handle potential division by zero (in case a day has no prompts)
modeMatrixPct(isnan(modeMatrixPct)) = 0;

% Plot stacked percentage bar chart
figure('Name','Chat Mode Usage per Day (Percentage)');
bar(days, modeMatrixPct, 'stacked');
xlabel('Day');
ylabel('Percentage of Prompts');
title('Chat Mode Usage per Day (Percentage)');
legend(modes, 'Location', 'northeastoutside');
grid on;

%% === Load metrics ===
metrics = logs(strcmp(logs.message, '/metrics'), :);

%% === Convert datetime day to numeric weekday 1=Monday ... 5=Friday ===
wday = weekday(metrics.day);
```

```
wday_adj = wday - 1;       % Monday=1 ... Sunday=0
wday_adj(wday_adj == 0) = 7; % Sunday=7

% Keep only Monday to Friday
validDaysIdx = wday_adj >= 1 & wday_adj <= 5;
metrics = metrics(validDaysIdx, :);
metrics.dayNum = wday_adj(validDaysIdx);

%% === Prepare full user-day grid (all users    days 1 to 5) ===
days = (1:5)';
[U, D] = ndgrid(allUsers, days);
combo = table;
combo.userId = reshape(U, [], 1);
combo.dayNum = reshape(D, [], 1);

%% === Group metrics by userId and dayNum ===
G = findgroups(metrics.userId, metrics.dayNum);
T_visits = table;
T_visits.userId = splitapply(@(x) x(1), metrics.userId, G);
T_visits.dayNum = splitapply(@(x) x(1), metrics.dayNum, G);
T_visits.PageVisits = splitapply(@numel, metrics.message, G);

%% === Outer join full grid with actual counts ===
T_full = outerjoin(combo, T_visits, ...
    'Keys', {'userId', 'dayNum'}, ...
    'MergeKeys', true);

% Replace missing visits with zero
T_full.PageVisits(isnan(T_full.PageVisits)) = 0;

%% === Pivot to wide format: one row per user, columns Day1...Day5 ===
T_wide = unstack(T_full, 'PageVisits', 'dayNum', 'VariableNamingRule', '
   preserve');

% Rename columns for clarity
dayCols = strcat("Day", string(days));
T_wide.Properties.VariableNames(2:end) = dayCols;

%% === Display result ===
disp('Page Visits per Day and User (Monday=1 to Friday=5):');
disp(T_wide);
```

```
% Load Excel data
file = 'Data_Pseudonym.xlsx';
sheet = 'Prompts';
data = readtable(file, 'Sheet', sheet);

% Extract input and output tokens
x = data.usage_numberOfInputTokens;
y = data.usage_numberOfOutputTokens;

% Filter out invalid entries
valid = ~isnan(x) & ~isnan(y) & x > 0 & y > 0;
x = x(valid);
y = y(valid);

% Sort by input tokens for smooth plotting
```

```
[x_sorted, idx] = sort(x);
y_sorted = y(idx);

% LOWESS smoothing to reveal real relationship
y_smooth = smooth(x_sorted, y_sorted, 0.05, 'lowess');  % 0.05 = smoothing span (adjust as needed)

% Reference line y = x
y_ref = x_sorted;

% Optional: Crop axes to the 95th percentile to avoid outliers
x_max = prctile(x, 95);
y_max = prctile(y, 94.83); % Skip one point to clean the graph image

% Plot
figure;
scatter(x, y, 10, 'filled', 'MarkerFaceAlpha', 0.3);
hold on;
plot(x_sorted, y_smooth, 'b-', 'LineWidth', 2);      % LOWESS trend
plot(x_sorted, y_ref, 'g--', 'LineWidth', 1.5);       % y = x reference

% Formatting
xlim([0, x_max]);
ylim([0, y_max]);
xlabel('Input Tokens');
ylabel('Output Tokens');
title('Smoothed Relation between Input and Output Tokens');
legend('Prompt Data', 'LOWESS Smoothed', 'y = x', 'Location', 'southeast');
grid on;
```

```
% Load Excel file and sheet
file = 'Data.xlsx';  % Adjust path if needed
sheet = 'Prompts';

% Read the table
data = readtable(file, 'Sheet', sheet, 'ReadVariableNames', true);
lengthsRaw = data.promptTextHistoryLengths;
numPrompts = height(data);

% Number of interpolation points (e.g., representing 0  100  %)
nInterp = 100;

% Matrix to store all normalized prompt curves
normalizedCurves = NaN(numPrompts, nInterp);

row = 1; % Row counter for valid prompts

for i = 1:numPrompts
    % Convert string to numeric array
    str = lengthsRaw{i};
    nums = sscanf(str, '%d,', Inf); % read comma-separated values
    if isempty(nums)
        parts = split(str, ',');
        nums = str2double(parts);
    end
    nums = nums(:)'; % ensure row vector

```

```
    % Skip if not enough points for interpolation
    if length(nums) < 2
        continue;
    end

    % Normalize x-values to range [0, 1] (prompt progress)
    steps = linspace(0, 1, length(nums));

    % Normalize y-values to final length (range [0, 1])
    finalLen = nums(end);
    if finalLen == 0 || any(isnan(nums))
        continue;
    end
    yNorm = nums / finalLen;

    % Interpolate to a fixed number of x-values (0  1   scale)
    xInterp = linspace(0, 1, nInterp);
    yInterp = interp1(steps, yNorm, xInterp, 'linear', 'extrap');

    % Store in matrix
    normalizedCurves(row, :) = yInterp;
    row = row + 1;
end

% Compute average curve across prompts
meanCurve = nanmean(normalizedCurves, 1);

% Smooth the average curve (optional)
smoothCurve = smooth(meanCurve, 0.2, 'loess'); % 0.2 = smoothing factor

% Plot
figure;
plot(linspace(0, 100, nInterp), smoothCurve * 100, 'LineWidth', 2);
xlabel('Prompt progress [%]');
ylabel('Text length [% of final length]');
title('Average normalized prompt growth curve');
grid on;

% Optional: show how many prompts were used
fprintf('Used prompts: %d of %d\n', row-1, numPrompts);
```

A.3.3 Project Documents from FHNW

Fachhochschule Nordwestschweiz

Windisch, 24.03.25

Project proposal for IP6, 25FS_IMVS24

Designing towards higher user awareness: UI strategies for reducing conversational AI energy consumption

Students: **Jack Gläser**
Simon Lüscher

Supervisors: **Martin Kropp**
Nitish Patkar

Stakeholder: Fachhochschule Nordwestschweiz FHNW

Project duration: 17.02.2025 - 14.08.2025

Version	Note	Autor
1	Initial version	Jack Gläser, Simon Lüscher
2	Draft	Jack Gläser, Simon Lüscher
3	Final	Jack Gläser, Simon Lüscher

1. Initial situation

Conversational AI consumes a lot of energy during its whole lifecycle. A significant part of this energy is required for the development and training of the model. But the actual conversational AI service consumes a great amount of energy as well. While a single prompt does not have a significant impact on the required energy it grows rapidly with a higher number of users. Over the period of a month the overall energy usage is already more than that of the training of the LLM. [1] The number of users is constantly increasing and already reached 987 million[2] in 2025. This adds even more significance to the inference part of the lifecycle of conversational AI since with a higher number of users the usage will increase even more.

Furthermore, the phase where users infer with the model also has an impact on previous lifecycle phases. If for example a user can be directed to prompt in a more optimized way the models could trained in an even more specific direction and with a less generic dataset. Ideally this would further reduce the energy consumption. [3] Although the effect size is not clear, and this should be further investigated it adds to the weight of the inference phase.

Another issue that we experienced ourselves and have seen with other people is that the trend is more to ask a conversational AI for an answer than using a search engine. Even for trivial questions. We hypothesize that users are not aware of the high energy consumption of a single inference with a conversational AI. Which is estimated to be at around 0.005 kWh per inference for GPT-4 for example. In comparison a search on google would only consume about 0.0003 kWh which is more than 16 times less.

[1] https://www.sciencedirect.com/science/article/pii/S2095809924002315#s0010
[2] https://www.demandsage.com/chatbot-statistics/
[3] https://www.sciencedirect.com/science/article/pii/S2095809924002315#s0010

2. Problem statement

The increasing adoption of conversational AI over traditional search engines is driving a significant rise in energy consumption. A 2023 study estimates that if users were to replace Google searches with interactions using large language models (LLMs), the energy demand for Google's AI alone would reach 29.3 TWh per year—equivalent to the electricity consumption of an entire country like Ireland. [4] This shift raises concerns about the sustainability of AI-powered search and the need for energy-efficient solutions.

Despite these implications, most users are unaware of the significant energy consumption associated with AI-driven interactions. Unlike traditional searches, which have well-optimized infrastructure with relatively low energy costs per query, LLM-based responses require substantial computational power. Since this energy usage remains largely invisible to end users, there is little public awareness or discussion on how AI adoption impacts global energy demand. This lack of awareness may slow down efforts to develop and implement more energy-efficient alternatives.

Furthermore, there is a lack of research specifically focusing on the energy consumption of AI inference in chatbots. While training LLMs is known to be highly energy-intensive, the long-term impact of frequent real-time inference remains underexplored. Companies developing these AI models do not transparently disclose their energy usage, making it difficult to assess the true environmental cost of conversational AI. This lack of data hinders informed decision-making and the development of more sustainable AI architectures.

3. Project vision

We strive to reduce the overall energy consumption associated with conversational AI by enhancing user awareness through targeted UI-based interventions. By researching and contributing to the state of the art, our project aims to understand how interface design alone can effectively influence user behavior regarding energy consumption. We will identify and evaluate UI features that increase user awareness, providing transparency and actionable insights into the energy impact of their interactions with conversational AI. Ultimately, our goal is to empower users to make informed decisions when and how to use conversational AI, fostering more sustainable and energy-efficient usage patterns without compromising user experience.

Research contributions:
We aim to present the state-of-the-art research on user awareness of the energy consumption associated with conversational AI. Additionally, we seek to conduct our own study to address existing knowledge gaps in this area.

Impact of UI functionalities on user awareness:
We want to research the potential impact that purely UI-based functionalities can/can't have in increasing user awareness and influencing user behavior to lower energy usage.

Increasing awareness in conversational AI:
Through implementing those functionalities and strategies, we intend to elevate overall user consciousness about their energy implications of their interactions with conversational AI.

[4] https://www.cell.com/joule/fulltext/S2542-4351(23)00365-3

4. Goals and research questions

4.1 Goals

1. Identify and evaluate methods to effectively measure user awareness of conversational AI energy consumption
2. Design and evaluate UI-only features that successfully enhance user awareness of energy consumption
3. Influence user behavior through increased awareness, resulting in decreased chatbot energy consumption.
4. Ensure all interventions maintain or enhance the user experience, preserving performance, responsiveness, and ease of use
5. (optional): Develop predictive capabilities to estimate a prompt's energy consumption based on user inputs and interactions.
6. (optional): Quantify potential savings (energy, tokens, API calls, CPU cycles) achievable through increased user awareness

4.2 Research questions:

A. To what extent are users currently aware of the energy implications associated with their chatbot interactions?
B. (Exploratory question): How can UI-based features most effectively increase user awareness regarding the energy consumption of conversational AI?
C. How strongly does increased user awareness correlate with reductions in conversational AI energy consumption?

5. Methodology

1. Literature Review

- Conduct a systematic review of existing **research** on:
  - Energy consumption in AI and conversational AI models.
  - Current strategies to enhance user awareness about energy consumption.
  - Existing UI-based interventions and their effectiveness in influencing user behavior.

2. Initial User Awareness Analysis

- **Survey**:
  - Develop and distribute a questionnaire to gauge users' current awareness of the energy impact of conversational AI.
  - Identify baseline awareness levels, knowledge gaps, and perceptions.

3. Design and Implementation of UI-based Interventions

- **Prototyping:**
  - Generate multiple UI design concepts aimed at increasing user awareness (e.g., visual energy indicators, impact scores, prompt suggestions…).
  - Prototype selected UI features, emphasizing transparency, ease of understanding, and actionability.
  - Implement selected UI interventions into a conversational AI environment, enabling real or simulated interactions.

4. Experimental Evaluation

- **User Testings:**
  - Designing and conducting a controlled experiment comparing user behavior with and without UI-based interventions.
  - Utilize metrics like the number of prompts, prompt length, frequency, and complexity of interactions to evaluate behavioral changes.
  - Assess UX impact through usability testing and user satisfaction surveys to ensure interventions do not degrade the conversational AI experience (performance, ease of use, responsiveness).

- **Energy Consumption Measurement:**
  - Deploy a monitoring system to measure and record energy usage, tokens consumed, API calls, and CPU cycles during interaction sessions.
  - Compare energy metrics before and after deploying UI-based interventions.
  - Use collected data to build and validate predictive models estimating the energy consumption of user prompts based on historical and behavioral data.

5. Data Analysis and Interpretation

- **Analysis:**
  - Perform statistical analysis to identify significant differences and relationships between user awareness, behavior, and energy consumption.
  - Analyze feedback from surveys and interviews using thematic analysis to identify qualitative insights into user experiences and perceptions of UI features.

6. Planning

We will follow an iterative approach throughout the project. Tasks will be created in GitLab and managed on a Board. [5] Our workflow is structured into two-week sprints, with a biweekly planning meeting to review the previous sprint and plan the next one. Coaches will receive updates either in person, if a meeting is scheduled, or via Teams in our dedicated channel. Apart from the planning meetings, all discussions and conversations will take place on Teams, with no additional meetings required.

The most important milestones and deadlines for the project are visible on GitLab[6] or in the following listing.

Deadlines:

Start project:	17.02
Finish project proposal:	18.03
Research, Literatur & Survey:	17.02– 30.03
Weg zur Thesis Workshop 1	15.03
Infrastructure setup:	15.03 to 31.03 (fully closed on 15.05 for Nitish)
Implementation phase:	01.04– 25.05
Weg zur Thesis Workshop 2	10.04 or 12.04
Experiment / User testings:	26.05– 08.06
Analysis of results from Experiment & Finish report:	09.06– 22.06
Weg zur Thesis Workshop 3	5.06 or 7.06
Sent first draft of report to Coaches:	22.06
Feedback and last adjustments:	23.06– 13.07
Hand-In Thesis:	~18.07
Theoretical Deadline:	14.08
Work on presentation & defence:	14.08 – presentation date (TBD)
Defence:	~ 01.09 bis 12.09

[5] https://gitlab.fhnw.ch/groups/25fs_imvs_ip6/-/boards
[6] https://gitlab.fhnw.ch/groups/25fs_imvs_ip6/-/milestones

7. Risiko Assessement

Risk	Level	Impact	Prevention / Strategy
Too high worklaod	Medium	Overloading of team members, loss of quality in work results	Iterative Approach
Absence of a team member	Low	Delays in project implementation, increased work pressure for remaining team member	Knowledge transfer through documentation of work processes and task distribution as well as reducing the scope.
Miscommunication in team	Low - Medium	Misunderstandings, inefficient collaboration	Regular meetings to clarify questions and ensure a smooth exchange of information
Insufficient communication with coaches	Low	Misunderstandings, unclear requirements, lack of support	Biweekly meetings and regular communication on Teams
Technical challenges	Medium	Delays in development	Read literature and support each other in learning new things, seek expert advice.
Time management problems	Medium	Delays in completion, increased time pressure	Realistic planning and stick to planned meetings and sprints
Topic is relatively new territory, finding research is difficult	Medium	Motivation drops, work stagnates	Take a break in case of blockages and ask your coaches for advice

Fachhochschule Nordwestschweiz

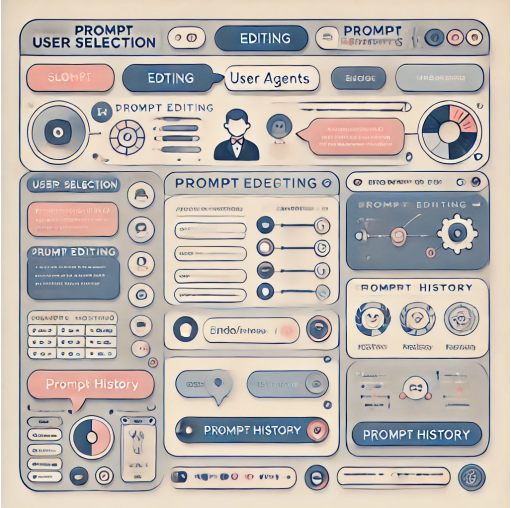
App Web | Design UX

25FS_IMVS24: Better ways to manage prompts in LLM Chatbots

			Priority 1	Priority 2
Advisor:	Martin Kropp Nitish Patkar	**Work scope:**	P5 oder P6	---
		Team size:	2er Team	---
Languages:	German or English			
Study course:	Computer Science			

Initial position

AI chatbots are widely used tools today for various routine purposes. Research shows that the UI and interaction design of most chatbots lack essential features, such as search functionality within history or prompt autocompletion. Additionally, each prompt contributes to an environmental toll, emphasizing the need for efficient prompt management. Currently, most chatbots handle past prompts in a similar, limited way, offering few options to revisit and branch from specific past prompts for conversations in new directions.



Better ways for prompt management

Objective

The overarching objective of this project is to advance the concept and implementation of our proof-of-concept AI chatbot. We aim to rethink how to make AI chatbots more user-friendly and improve the overall user experience. Specifically, within IP5, we want to explore interaction mechanisms that enable users to work effectively with past prompts, reducing the need for repeated re-prompting. This involves experimenting with visualization alternatives, addressing design and implementation challenges, and validating the effectiveness of the developed solutions. For IP6, students can also explore prompt speculation- speculating next prompt based on previous interaction.

Problem statement

In today's prompt management for LLMs, UI/UX faces several challenges. Usability suffers due to complex prompt tuning and the lack of intuitive interfaces for users to refine prompts without technical expertise. Traceability is limited, as it's often unclear how specific prompts influence responses, complicating debugging and optimization efforts. Additionally, there is a lack of historicization tools to track prompt versions, changes over time, and their impacts on outputs, which hinders iterative improvements and makes it difficult to maintain consistent performance across updates. Together, these issues make prompt management cumbersome and less accessible for users..

The following questions should be answered:

- What features do current popular AI chatbots support for interaction and prompt management?
- What visualization and interaction mechanisms can enhance interaction with past prompts?

Technologies/Technical emphasis/References

Web development, preferably Python, Angular, SQLite, to be discussed.

Note

There has been earlier work on a this https://nitishspatkar.github.io/pdfs/IP5_FS24_Simon_Jack.pdf which can be used as inspiration by the students.