Project Safety and Reflection Report

Laura Kotalczyk[a] and Manuel Mühlberger[a]

1. Safety of GenAI

1.1. Identified Risks

With the current setup of our service, we see two distinct areas bearing potential risks, namely, data privacy and security. The following paragraphs discuss these aspects in more detail.

1.1.1. Data Privacy Concerns

Since user data is a primary asset of our service, data privacy, and compliance with GDPR in particular, is a central part of our risk management. We store three main types of information: user profiles (containing names, physical metrics, and medical conditions), as well as audio and image files recorded by our users. Under GDPR, these categories are classified as personal data, with medical information specifically treated as highly sensitive. Given this classification, we prioritize the secure storage of this data and implement strict access controls to prevent any unauthorized disclosure.

1.1.2. Security Concerns

Since our service relies on a server-client architecture and utilizes an API to connect with a Vision Language Model (VLM) for nutrient estimation, the application is naturally exposed to risks such as Denial-of-Service (DoS) attacks and API exploitation. This vulnerability is especially significant because our meal-logging feature relies on a credit-based system for VLM access. Because of this setup, exceeding rate limits would lead to more than just service downtime. Without proper safeguards, these surges in API calls would cause direct financial losses in addition to the disruption of our service.

1.2. Risk Mitigations

1.2.1. Data Privacy

In the context of Risk Mitigations regarding Data Privacy, the following measures have been implemented. To further enhance security and privacy, user data is managed within an on-premise database, while all external requests to the Vision Language Model (VLM) are sent through a single, unified account. By routing all interactions as coming from one “outer user” regardless of the underlying user count, we make it significantly more difficult for external providers to perform profiling on individual users. Furthermore, to protect sensitive audio recordings, we utilize a locally hosted Whisper API for initial processing. This ensures that only the resulting text transcription is shared with the VLM, while the original audio files remain secured within our local infrastructure.

1.2.2. Security

To mitigate potential attack vectors concerning service availability, several mechanisms were implemented. First of all, we established a user authentication mechanism based on JSON Web Tokens

[*]

(JWT) and certificates, ensuring that only authenticated users can interact with our service and submit requests to the VLM for meal nutrient estimation. To further harden this system, user passwords are securely hashed and salted before storage, and TLS encryption is enforced for all client-server communication to protect data in transit. This approach prevents direct public access to the VLM request endpoint, leveraging the server as a secure gateway for authenticated requests.

To manage resource consumption and mitigate abuse, we implemented a multi-tiered rate-limiting strategy. During the development phase, we integrated a per-user limit — currently set to five requests daily — which can be easily scaled for production. Additionally, we enforced a global server-side limit on requests per minute to protect the system from broad automated misuse. To maintain data consistency, we enforced a strict output format. The model is required to return a specific JSON object rather than raw text. This serves as a validation layer: the client expects only this format and will reject any response that does not conform to the schema, which helps mitigate the impact of unexpected or malformed model outputs. Finally, the infrastructure is protected by isolating services into Docker containers and using a proxy as a load balancer and firewall to defend against Denial-of-Service (DoS) attacks.

2. Lessons Learned and Reflections

Throughout the project, we gained significant insights into project and team management as well as architectural design, and the usage of GenAI, that are further detailed in the following sections.

2.1. Team and Project Management

Throughout the project, we gained significant insights into project and team management. In particular, regular in-person meetings proved highly effective for maintaining alignment, sharing progress, and quickly agreeing on next steps. At the same time, we learned that differences in prior experience and expectations can make task scoping and ownership more challenging than anticipated. For future projects, we would place more emphasis on forming teams with a more consistent baseline in technical experience and engagement, to enable a smoother distribution of responsibilities and more predictable execution.

2.2. Architectural Design

On the architectural side, we found that an initially "complex-looking" setup can still be a pragmatic and robust choice in practice. While the overall system design — especially the Docker-based service separation combined with the Pangolin tunnel and proxy layer — required some upfront effort to configure and understand, it proved very reliable during development and testing. Similarly, integrating Whisper and operating with a self-signed certificate worked without issues and did not become an operational bottleneck. Potential improvements mainly relate to production-readiness and trust assumptions.

First, relying on externally hosted VLM providers introduces uncertainty regarding data handling and privacy practices. A locally hosted VLM would reduce this dependency, but was not feasible in our case due to hardware constraints. Second, for a public deployment, replacing the self-generated certificate with a universally trusted certificate (e.g., via Let's Encrypt) would improve usability and avoid trust warnings on the client side.

2.3. Insights on GenAI and its Applications

When we began the project, we expected that the top-ranked models on public benchmarks and the most precise prompts would deliver the best results. However, our findings challenged the assumption that complexity yields superior performance, demonstrating that simpler methods often outperform more complex configurations in practical, real-world scenarios. We observed that high benchmark scores are not always indicative of real-world performance, as the effectiveness of GenAI is heavily context-dependent. Consequently, achieving the optimal configuration requires a systematic, experimental approach, adjusting prompts and parameters to identify the most impactful variables. While established techniques provide a foundation for improvement, the 'black-box' nature of AI as well as the wide landscape of models, and corresponding large number of tunable parameters, make this a tedious and time-consuming process, requiring significant iteration to reach high-quality results. Furthermore, the project provided practical evidence that data is the most critical asset in AI systems. Beyond the vast datasets required to train the off-the-shelf models we utilized, the availability of high-quality 'ground truth' data was essential for validating performance and conducting comparisons between different configurations and models. In our specific use case, curating representative, non-skewed and realistic meal data was challenging and time-intensive. The process involved preprocessing data into our input format, (image, transcription) pairs, and mapping them to a JSON-based ground truth file containing nutritional metrics such as calories, carbohydrates, proteins, and fats. These ground truth meals allowed us to benchmark our service's accuracy against competitor apps and evaluate how our performance compares to the current market.

A key takeaway from this project also was that the integration of generative AI into a consumer-facing application proved more complex than initially anticipated. Beyond the basic implementation, significant consideration had to be given to the underlying architecture required to orchestrate API requests and manage data flows. Moreover, embedding AI services necessitates a robust framework for data privacy and security, as addressing these factors is a prerequisite for ensuring GDPR compliance and developing a responsible, trustworthy user experience.

3. Acknowledgments of GenAI Usage

Generative AI was used in the process of writing this document in terms of improving the vocabulary used.