EatNeat

Team **SixSeven**

Arin Bandyopadhyay · Chinmay Pankaj Torvi · Himanshu · Taes Padhihary
Udgam 2026 · https://github.com/maiHydrogen/Encode

---

1. Summary of EatNeat

While looking up nutritional details about a product, the modern consumer is overwhelmed by complex ingredient lists, deceptive marketing, and scientific jargon made to deter the consumer. Traditional apps require users to manually input data or scan specific barcodes against limited databases. EatNeat flips this model using Generative AI, making nutritional information more accessible for the consumers.

A simple snap of a box's back or a simple query of a couple words is all you need to get the treat examined by EatNeat. EatNeat will provide you with the goods and bads of anything on demand, based on just a small text query or a single snap.

EatNeat's inference based model also allows for users to put in local names for dishes, ingredients and cuisines, and get a response in their own language(if preferred), and it also allows for sidestepping of spelling errors, preventing the shoehorning of users into writing neat English.

2. Information Gap

Key Problems

In the current market, consumers face three friction points:

1. **Cognitive Overload:** Ingredient labels are written in a most unreadable format, obscuring relevant information.
2. **Database Limitations:** If a product is not in the database, the app fails.
3. **Lack of Context:** Direct facts like "a product contains x grams sugar" is not helpful to the average consumer.

EatNeat provides the challenges by providing the relevant information in a simpler and more understandable way. For example, instead of "x grams of sugar" it says "*contains higher amount of sugar than healthy*", "*eat only as a treat*" etc.

We leverage Multimodal AI (Text, Vision, and Voice) to reason about food just like a human expert would, but instantly. It provides the goods and bads of any product in a neat list manner, which is easy to understand.

3. How EatNeat solves the problem

3.1. Core Features

- **Instant Vision:** Users take a picture of **any** package (front or back). The AI reads the text visually (OCR-free) and identifies the product.
- **Voice-First Accessibility:** Recognizing that typing on mobile is tedious, we integrated Voice Search. Users can simply ask, *"Is this safe for diabetics?"*
- **Data Privacy:** A key selling point. We do not store user photos, chat history, or profiles. The analysis happens in a "state-less" environment, building immediate trust. The output is immediate, and the input is thereafter forgotten.

3.2. The Logic Flow

Unlike traditional apps that look up rows in a spreadsheet, EatNeat uses the flow in **Figure 2**.

The architecture is designed to be **stateless**. No user database is required, ensuring zero data retention and maximum privacy.

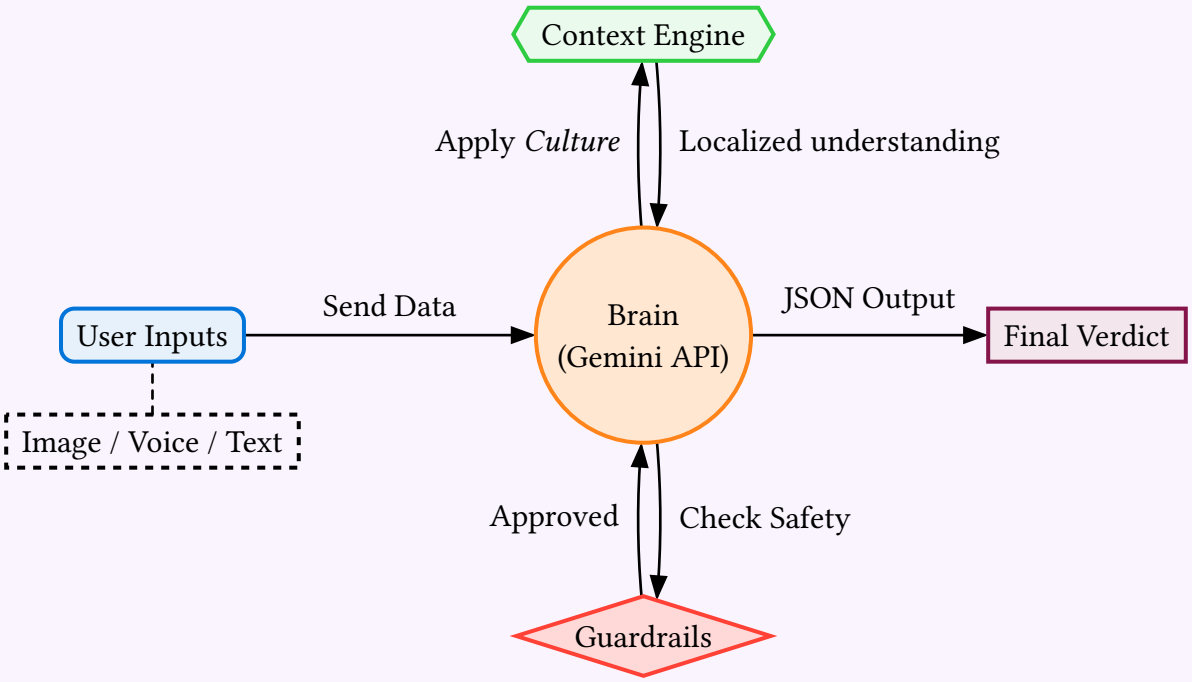

Figure 2: EatNeat Flow

4. Future Improvements

Next Steps

- Move on from Gemini API to a self-trained, fine-tuned and hosted LLM.
- Add new features like adding a new tab for similar foods with similar benefits.
- Increasing and optimising the database for not only Indian but worldwide queries, with more language support.
- Add drag & drop features for a better user experience on laptops/desktops.
- Figure out a way to monetize this product.

5. Key Innovations & Strategic Adjustments

To ensure market fit in India and robust safety, we implemented specific Hindi/Hinglish replies for Hindi queries and made a robust inference system for it as well.

5.1. The "India-First" Context Layer

Generic AIs often applaud Western diets but fail to understand Indian nuances. We engineered the prompts to recognize:

- **"Hinglish" Support:** The AI understands queries like *"Kya ye healthy hai?"* and ingredient names like *"Jeera"* or *"Maida."*
- **Cultural Diet Norms:** It flags "Maida" (Refined Flour) as a negative indicator and recognizes "Ghee" as a positive fat source, differentiating it from generic saturated fats.

5.2. Brand Safety & Sanitization

A major risk with GenAI is hallucination or misuse. We built a strict filtering layer:

- **Typo Correction:** It aggressively fixes food typos (e.g., "Maggee" → "Maggi").
- **Anti-Abuse Filter:** The system detects non-food slang or NSFW terms (e.g., misreading text as inappropriate slang) and returns a "System Error" rather than attempting to analyze it as food. This protects the brand reputation. Small unfortunate typos are inferred as the closest food items.

5.3. Generative UI

Instead of a static screen, the UI adapts based on the verdict:

- **Green Theme:** For healthy items (Positive reinforcement).
- **Red/Yellow Theme:** For warnings (Cautionary visuals).
- **Pros & Cons:** Dynamically generated bullet points based on the specific product, not a generic list. It gives the points based on user input.

6. Conclusion

EatNeat represents the shift from "Data-Heavy" apps to "Intelligence-First" experiences. By solving the friction of manual entry and adding a layer of cultural intelligence and safety, we have created a product that doesn't just display data—it helps users make better decisions instantly.

Eatneat's utmost priority is the comfort of the user, both while asking their queries and receiving the answers required.