Vyb AI Assignment

🏁 Goal: Can You Build a Smart Nutrition Estimator?

At VYB, we're making India's smartest food brain. Given how home-cooked Indian meals vary wildly, your challenge is to create a **resilient system** that:

> 🔍 "Given a home-cooked dish name, estimate its nutrition per serving, even if the data is partial, ambiguous, or broken."

📦 You'll Get:

Nutrition Database** (per 100g) for common ingredients. Here is the link to it
➕ Assignment Inputs
Household Measurement Reference
Test Dishes JSON** with real-world messiness (see below)

🎯 Your Challenge

Build a program (Node.js or Python preferred) that does:

✅ Part 1: Core Pipeline

1. **Input:** Dish name (e.g., *Paneer Butter Masala*)
2. **Fetch ingredients** (simulate or use LLM if needed)
3. **Convert ingredient units** to household measurements
4. **Map ingredients to the Nutrition DB** (handle synonyms, spelling variants, etc.)
5. **Estimate grams** using measurement mapping
6. **Calculate total nutrition** (per 100g basis, then scale)
7. **Identify dish type** from a predefined list (e.g., Wet Sabzi)
8. **Output nutrition per 1 katori**

🧪 Part 2: Messy Reality (Edge Case Handling)

You will be given a test JSON with 5 dish inputs like:

```json
[
  { "dish": "Jeera Aloo (mild fried)", "issues": ["ingredient synonym", "quantity missing"] },
  { "dish": "Gobhi Sabzi", "issues": ["ambiguous dish type"] },
  { "dish": "Chana masala", "issues": ["missing ingredient in nutrition DB"] },
  { "dish": "Paneer Curry with capsicum", "issues": ["unit in 'glass'", "spelling variation"] },
  { "dish": "Mixed veg", "issues": ["no fixed recipe", "ambiguous serving size"] }
]
```

⛏️ **Your Job:**

* Show how your code handles each (default values, logging, graceful degradation)
* Include a **log of assumptions made per dish**

🧠 Part 3: Reasoning Task

Manually solve **1 of these**:

* Map "lightly roasted jeera powder" to a nutrition entry, and explain why.
* Dish weight is 700g cooked, and raw ingredient weight totals 950g. What's the loss ratio?

Separately, adjust the final nutrient values to a standard 180g serving size, assuming total nutrition was calculated for the full cooked quantity..

 Bonus: Do this **without using ChatGPT.**

🧹 Bonus

* Add a **unit test** for your quantity conversion function (e.g., tbsp to grams). Make sure your function accounts for ingredient-specific density — 1 tbsp of oil is \~13g while 1 tbsp of sugar is \~12.5g. Test at least 3 different ingredients with varying densities.
* Create a CLI or simple Express API that accepts dish name and gives output.

📜 Deliverables

* `script.js` or `main.py`

* `readme.md`: your approach, assumptions, fallback design
* `debug-log.txt`: how you handled each of the 5 test dishes
* Output JSON for each test case (input → final nutrition estimate)
* Deploy & share a link to see working version

📊 Evaluation Criteria

Area	Weight
Correctness & Accuracy	35%
Code Logic & Modularity	25%
Real-World Handling & Logging	20%
Manual Reasoning & Edge Case	10%
Documentation & Clarity	10%

⚠️ Anti-Plagiarism Notice

We *encourage* using LLMs for fetching recipes, but not for **manual reasoning or debugging**. Your real skills matter.
Shortlisted candidates will be asked to:

* Join a 30-minute live walkthrough or
* Do a mini bug-fix round based on your submission.

🚀 Deadline

Submit within **48 hours** of receiving this pack.