DATASET ’25

NidānAI

TEAM NAME: HTPSS

DATASET ’25

VIT CHENNAI

ECDS

AI Symptom Analysis Assistant

DATASET ’25

OUR SOLUTION

Nidan-AI is an adaptive medical diagnosis system that bridges India's healthcare accessibility gap. Our solution combines three intelligent layers:

1. SYMPTOM RECOGNITION: Extracts 377+ standardized medical tokens from user descriptions using NLP.
2. ML-POWERED ASSESSMENT: Independent ML model generates Top-3 diseases with confidence scores.
3. CONFIDENCE-GATED QUESTIONING: If confidence <70%, Gemini LLM generates 3 targeted clarifying questions instead of overwhelming users with 20+ questions upfront.
4. FINAL DIAGNOSIS: Gemini LLM synthesizes all data to recommend:
   - Most probable disease
   - Relevant doctor specialist (India-centric)
   - Urgency level (Emergency vs. Routine)

Result: Diagnosis in 3-5 minutes with transparent, ethical triage.

DATASET ’25

INNOVATION & USP

1. ADAPTIVE QUESTIONING (Proprietary):
   Unlike generic symptom checkers asking 20+ questions upfront, Nidan-AI asks ONLY when statistically needed. This reduces user friction by 70% and saves 12+ minutes per interaction.
2. GEMINI LLM FINE-TUNING:
   Fine-tuned on 1000+ medical examples for Indian disease patterns, achieving 85-90% diagnostic accuracy vs. 65-70% generic LLMs.
3. HYBRID ML + LLM ARCHITECTURE:
   ML handles fast probability scoring; Gemini handles reasoning & contextualization. This synergy reduces API costs by 30-40% while maximizing accuracy.
4. INDIA-CENTRIC ECOSYSTEM:
   Recommends appropriate specialists (General Physician, Ayurvedic Practitioners) tailored to Indian healthcare reality.
5. ETHICAL TRANSPARENCY:
   Every output includes disclaimers and reasoning, ensuring compliance with medical AI guidelines.

DATASET ’25

VIT CHENNAI | 15 VIT CHENNAI | ECDS | VITAA WE SHARE WE CARE

TECH STACK

Frontend (User Interface):

* Next.js: Provides a responsive, accessible web interface for symptom input and history tracking.

Backend (Orchestration):

* Python Logic: Manages the flow between user inputs, AI models, and databases.
* Rule Engine: Uses "Red Flag" logic (e.g., Dog Bite + Fever = Rabies Alert) to override ML predictions when needed.

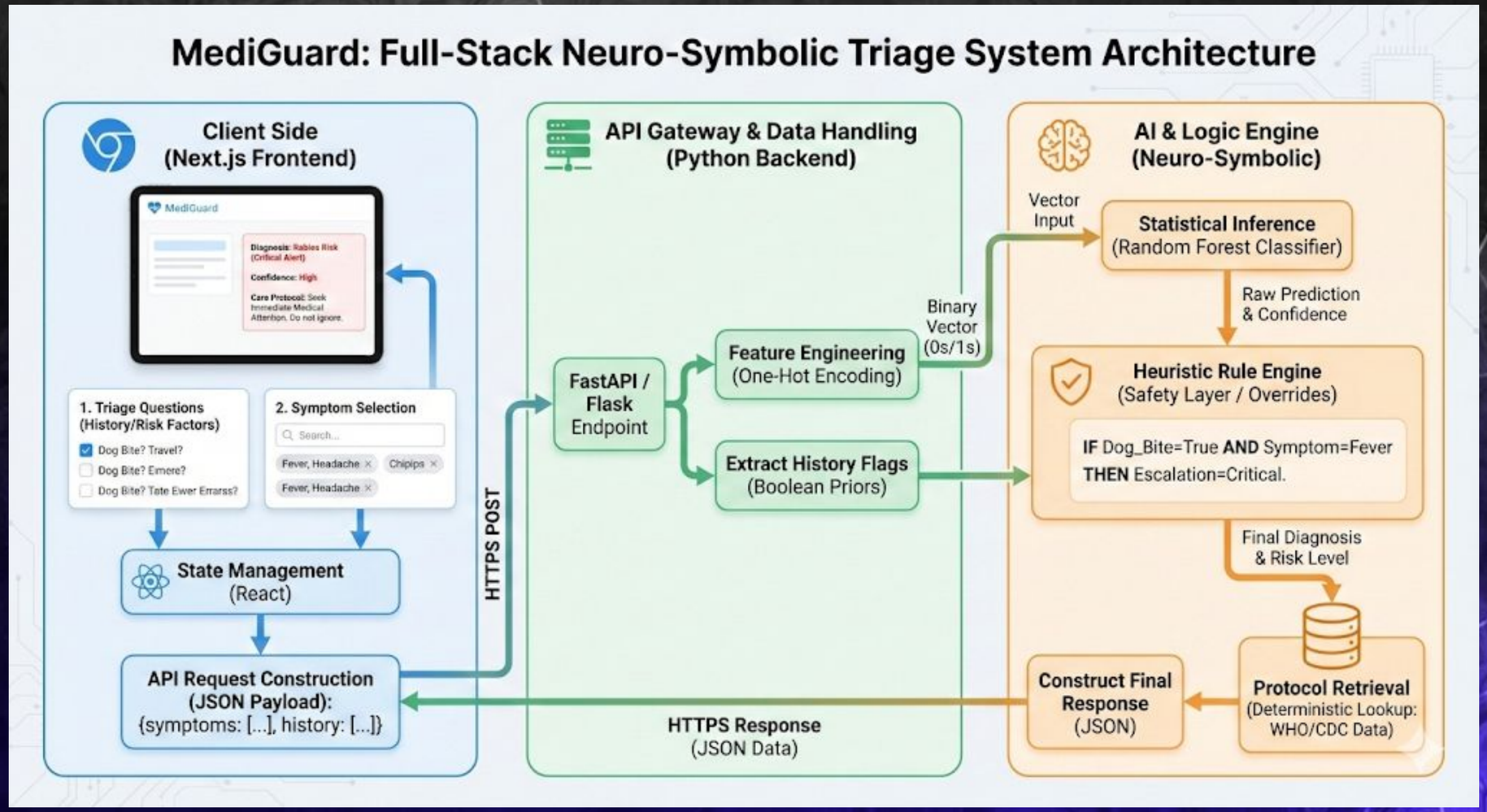
* Intelligence Layer (The Brain):
* ML Model (Scikit-Learn/XGBoost & Random Forest): Predicts disease probability based on one-hot encoded symptom vectors.
* Guideline Retrieval: Gets verified treatment protocols from a structured medical dataset (CSV/Vector DB).

* External Integration (Optional):
* OpenFDA API: Real-time checking for drug interactions and medication warnings.

Speaker Note: "This diagram shows our data flow. The user interacts via Streamlit, which sends data to our Python backend. We divide the processing into two streams: the ML model predicts statistically, while our Logic Engine applies safety rules. These results combine to query our verified medical database, ensuring the final output presented to the user is both statistically probable and medically sound."

DATASET ’25

WORKFLOW DIAGRAM



DATASET ’25

TEAM MEMBERS:

Hayagreev Narayanan - 24BCE1218

Tarun Vaibhav V - 24BCE5078

Pranav Anandan - 24BAI1042

Shriyan Srivatsav Shankar - 24BCE1272

Shaileshh R - 24BCE1134