Hackathon Strategy for MERN Developers (No Paid LLMs)

PS 2 – Mapping User Queries to Categories (Tredence)

Objective: Build a system that maps user queries to a product hierarchy without relying on paid APIs.

- Dataset Preparation: Use the training dataset provided to build a simple ML/NLP model.
- Backend (Node/Express): Implement a classification API that takes queries and maps them to categories.
- Model Options: Use TF-IDF + Cosine Similarity or a Naive Bayes classifier (via Python microservice or Node NLP libraries).
- Database (MongoDB): Store product hierarchy and user queries for evaluation and analytics.
- Frontend (React): Build a dashboard where users can input queries and view mapped categories.
- Demo Enhancements: Show classification accuracy, allow query testing with results visualization.

PS 5 – Health Report Analysis & Insight Engine (FOXO)

Objective: Build a rule-based system that analyzes structured health reports and provides insights.

- Dataset: Assume reports are structured (CSV/JSON with biomarker values). Avoid unstructured PDFs unless clarified.
- Backend (Node/Express): API to accept report uploads, parse structured data, and run rule-based checks.
- Rules Engine: Define thresholds (e.g., cholesterol > X = High). Store thresholds in MongoDB for flexibility.
- Insights: Provide normal/abnormal status, correlations (e.g., high sugar + high BMI = diabetes risk), and basic recommendations.
- Frontend (React): Upload report interface + dashboard with results, graphs (using chart libraries).
- Demo Enhancements: Add predictive trend analysis (simple linear rules, no ML needed).

PS 6 – Smart Food Recommendation System (FOXO)

Objective: Build a recommendation engine that suggests food items based on health conditions without LLMs.

- Dataset Preparation: Create or use a dataset mapping diseases to food items (good/bad).
- Backend (Node/Express): API that takes user condition and returns food recommendations with explanations.
- Database (MongoDB): Store food-condition mappings with metadata (nutritional values, reasoning).
- Frontend (React): Search bar + chat-style UI where users ask 'Is mango good for diabetes?' and get Yes/No with explanation.
- Logic: Use simple string matching/fuzzy matching to process queries without LLMs.
- Demo Enhancements: Add filters (e.g., vegetarian, calorie limits), and show related health benefits.