Fynd AI Intern – Take Home Assessment

Name: Akshay Bhadule
Assignment: AI Intern Take Home Assessment

🔹 Objective

The objective of Task 1 is to evaluate how prompt design impacts a Large Language Model's ability to predict star ratings (1–5) from customer review text.

🔹 Dataset Used

A real-world dataset (yelp_reviews.csv) containing customer reviews and corresponding star ratings was used.
A random subset of approximately 200 reviews was selected to efficiently test prompt performance while maintaining data diversity.

🔹 Prompt Design

Three prompt versions (V1, V2, V3) were designed with increasing levels of structure and constraints.
Each prompt asked the model to predict a rating and return the output strictly in JSON format.

🔹 Observations

As prompt instructions became more explicit and structured, the model's outputs became more consistent and reliable.
Prompt V3 produced the most stable and well-formatted responses.

Task 2: AI-Powered Review Management System

🔹 Objective

The objective of Task 2 was to build an end-to-end system that accepts user reviews, processes them using an AI service, stores structured outputs, and provides an admin interface for monitoring submissions.

🔹 System Overview

The system consists of a FastAPI backend, a SQLite database, and two user interfaces:
a user-facing review submission page and an admin dashboard with auto-refresh functionality.

🔹 AI Processing

For each submitted review, the system attempts to generate an AI-based summary, suggested action, and response.
If the AI service is unavailable, the system gracefully falls back to a default response while still storing the review data.

🔹 Error Handling & Design Decisions

The system is designed to remain stable even when external AI services fail.
Reviews are always stored in the database, ensuring no data loss.
This approach reflects real-world production considerations where external dependencies may be unreliable.

🔹 Conclusion

This assignment demonstrates prompt engineering, API design, error handling, and full-stack integration using modern tools.
The final solution prioritizes clarity, robustness, and maintainability.