Final Requirement in ITEP 308 – System Integration and Architecture I First Semester, Academic Year 2025-2026

Members:

Espinile,Ralph Ryan G.

Melodillar,Lawrence Angelo B.

Torres,Aprilene E.

ReviewMood

Introduction:

ReviewMood is a web application that analyzes customer reviews using sentiment analysis. Its goal is to help businesses quickly understand whether customer feedback is positive, neutral, or negative. The system processes customer text reviews, runs them through a machine learning sentiment model, then displays the result to the user.It also shows a simple and clean output so users can instantly see the mood or attitude of customers based on their written feedback.

Problem Definition:

Businesses receive many customer reviews, but manually checking each review takes time and can lead to missed issues, trends, or complaints. There is no instant way for businesses to see whether overall customer mood is positive or negative.

Purpose of the Application:

ReviewMood automatically analyzes customer review text using sentiment analysis. It classifies reviews as positive, neutral, or negative, helping businesses understand customer satisfaction quickly and make better decisions.

Design thinking

Hills: What the system enables users to do

Hills	Statement Hills	Success Measure

1	Users can input customer reviews and instantly see sentiment results.	90% of reviews processed correctly with sentiment accurately identified.
2	Users can track and review mood trends through organized summaries.	Users can generate weekly/monthly trend summaries within 10 seconds, with 95% accuracy.
3	The system simplifies understanding customer satisfaction with clear output.	85% of users report that the output is easy to understand in surveys or feedback forms.

Sponsor User: The representative primary user

Business owners or **customer service teams** who need fast insights from customer reviews without reading each one manually.

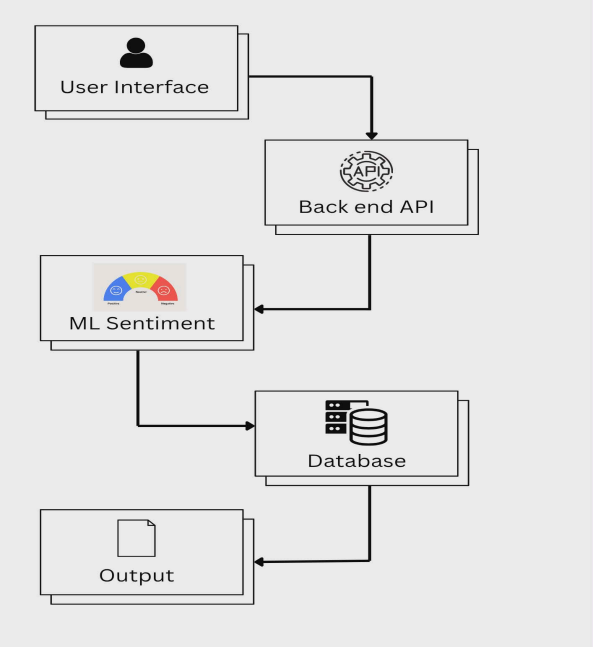
Playback: What feedback you gathered and how you improved the design

Problem	**Solution**
Users wanted faster results	System optimized for quick sentiment output.
Users wanted clearer feedback display	Improved layout and labels.
Users wanted better accuracy sentiment model upgraded for more precise classifications.	sentiment model upgraded for more precise classifications.

System Architecture

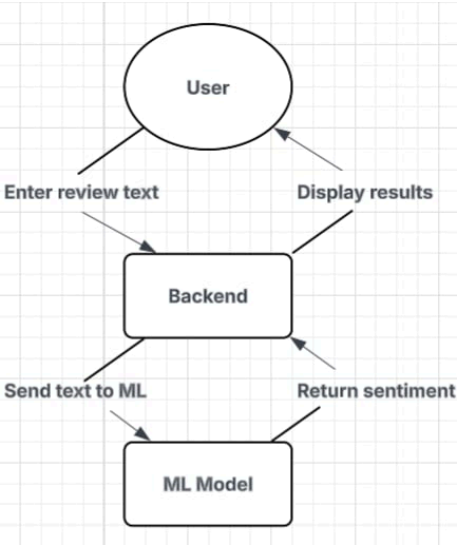
Architecture Diagram (Text)

User Interface → PHP Backend → ML Sentiment Model → Results Output



Data Flow

1. User inputs customer review text.
2. The backend sends text to the ML model.
3. ML model returns sentiment classification.
4. Results are displayed to the user.



Libraries Used

GuzzleHTTP – HTTP requests

PSR-7, PSR-17, PSR-18 – HTTP message standards

Symfony Components – utility functions

Ralouphie/getallheaders – header parsing

ML Model Integration (backend calls this for sentiment prediction)

ML Integration

The ML model performs:

Sentiment classification

Polarity detection (positive, neutral, negative)

Confidence scoring

System Demonstration

Key Features

- Enter a customer review text
- Automatic sentiment analysis
- Quick display of mood result
- Clean and simple output

ML Functionality

- Uses NLP to interpret customer comments
- Analyzes tone, keywords, and context
- Outputs sentiment category and optional score

Input → Process → Output

Input: Customer types or submits a review

Process: System sends it to ML → model analyzes sentiment

Output: Displays "Positive," "Neutral," or "Negative" with details