Project Charter:
AI-Powered Sentiment Analysis & Recommendation System

1. Project Title

AI-Powered Sentiment Analysis and Recommendation System for E-Commerce Reviews

2. Project Purpose

The objective is to develop a practical, lightweight machine learning-based system for e-commerce product review analysis, sentiment classification, key product aspect extraction, and personalized product recommendations. This system will apply ML, full-stack development, and project management in a structured fashion.

3. Project Objectives

- Put ML theory into practice: building sentiment and aspect extraction models with Python and Scikit-learn.
- Full application system with an ML backend, database, and Django web interface.
- Strengthen skills through design, implementation, testing, and documentation.

4. Scope Overview

- Text preprocessing, and multilingual handling-EN/ES
- Sentiment classification (positive/neutral/negative)
- Simple aspect extraction: quality, shipping, price, etc.
- Lightweight collaborative/content-based recommendation engine
- Django web interface + MySQL database
- Documentation, UML Diagrams, and Testing

5. Deliverables

- Cleaned and processed dataset
- Trained ML models and evaluation metrics
- Aspect extraction module
- Recommendation engine
- Django web application: review input, results, recommendations
- MySQL database with integrated pipelines

- Project report, UML diagrams, tests, and final demo

6. Assumptions

- Dataset remains static - no continuous data ingestion.
- Models will be run locally or on a single machine.
- Limited multilingual support, including only English and Spanish.

7. Constraints

- Limited development time
- Lightweight models only
- scikit-learn, Django, MySQL must be used.

8. Original Risks

- Imbalanced Data Affecting Sentiment Accuracy
- Low Spanish accuracy due to fewer samples
- Recommendation engine sparsity
- Integration issues between ML pipeline and Django

9. Success Criteria

- Model achieves reasonable precision
- Fully functional Django app demonstrating end-to-end flow
- Clean documentation and test coverage