**Sentiment Analysis Project – Actionable Summary**

**📌 Objective**

To develop a sentiment analysis model that classifies user-generated text (e.g., app reviews) into **Positive**, **Neutral**, or **Negative** sentiments, using Natural Language Processing (NLP) and Machine Learning.

**🔧 Tools & Technologies Used**

* **Programming Language**: Python
* **Libraries**: Pandas, Scikit-learn, NLTK, Matplotlib, Seaborn, WordCloud
* **ML Algorithm**: Naive Bayes Classifier
* **NLP Techniques**: Text Cleaning, Tokenization, Lemmatization, Stopword Removal, TF-IDF

**📊 Steps Performed**

1. **Dataset Preparation**
   * Created a sample dataset of 10 text reviews with labeled sentiments (Positive, Neutral, Negative).
2. **Data Preprocessing**
   * Removed special characters, lowercased text, removed stopwords, and applied lemmatization.
3. **Feature Extraction**
   * Used **TF-IDF Vectorizer** to convert text into numerical features for model training.
4. **Model Building**
   * Trained a **Multinomial Naive Bayes** classifier on the preprocessed data.
5. **Model Evaluation**
   * Evaluated using **Confusion Matrix** and **Classification Report** (Precision, Recall, F1-Score).
6. **Visualization**
   * Displayed **Word Clouds** for each sentiment class.
   * Visualized model performance using a heatmap of the confusion matrix.

**📈 Results**

* Successfully classified reviews into three sentiment categories.
* Demonstrated foundational understanding of NLP pipelines and text classification.
* Achieved high interpretability using simple models and clear visual outputs.