## Project Overview: Grammar Scoring Engine for Voice Samples

**1. Project Goal**

The primary objective of this project is to develop a Grammar Scoring Engine designed to analyze voice samples. The engine accepts audio files as input, performs feature extraction to capture relevant speech characteristics, and employs machine learning models to predict a grammar score, representing the grammatical correctness of the spoken content within the audio sample.

**2. Used Key Components**

* **Input:** Audio files containing voice samples.
* **Audio Processing & Feature Extraction:** Utilize libraries like **Librosa** to load audio files and extract meaningful acoustic features. Examples of extracted features include Mel-Frequency Cepstral Coefficients (MFCCs), chroma features, and spectral contrast. **Soundfile** might also be used for basic audio file input/output operations.
* **Data Handling and Manipulation:** Employ **Pandas** to organize the extracted features alongside their corresponding grammar scores (likely loaded from a separate source, e.g., a CSV file) into structured DataFrames. **NumPy** is used for underlying numerical computations and array management.
* **Machine Learning Model Training:**
  + **Data Splitting:** Use **Scikit-learn**'s train\_test\_split function (sklearn.model\_selection) to divide the dataset into training and validation sets. This allows for model training and unbiased performance evaluation.
  + **Model Selection:** Implement and compare various regression models from **Scikit-learn** (sklearn.linear\_model):
    - **Linear Regression:** Serves as a baseline model for predicting grammar scores based on features.
* **Model Evaluation:** Assess the performance of the trained models using standard regression metrics from **Scikit-learn** (sklearn.metrics), such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared (Coefficient of Determination). 1
* **Data Visualization:** Use **Matplotlib.pyplot** to create visualizations for:
  + Understanding the distribution of grammar scores (e.g., histograms, boxplots).
  + Potentially exploring relationships between extracted audio features and the target grammar score.

**3. Libraries Utilized**

* **Data Handling and Manipulation:**
  + **Pandas:** For efficient data loading, manipulation, and analysis in tabular structures (DataFrames).
  + **NumPy:** For fundamental numerical operations, especially array handling.
* **Audio Processing:**
  + **Librosa:** For loading audio files and extracting a wide range of acoustic features relevant to speech analysis.
  + **Soundfile:** (Potentially) For reading and writing audio file formats.
* **Machine Learning:**
  + **Scikit-learn (sklearn):** A comprehensive suite for machine learning tasks. Key modules used include:
    - sklearn.model\_selection: For data splitting (train\_test\_split) and hyperparameter optimization (GridSearchCV).
    - sklearn.linear\_model: Provides implementations of linear models (Linear Regression, Ridge, Lasso, Elastic Net).
    - sklearn.preprocessing: (Potentially used for data scaling, though not explicitly mentioned in current implementation details).
    - sklearn.metrics: For calculating model evaluation metrics (MSE, RMSE, MAE, R-squared).
* **Visualization:**
  + **Matplotlib.pyplot:** For generating static plots and visualizations to understand data distributions and model results.