**Prototype Documentation:**

**Approach:**

1. **Emotion Detection:**
   * Used MFCCs and chroma features for emotion detection.
   * Trained a RandomForestClassifier for emotion classification.
2. **Speaker Diarization:**
   * Integrated a placeholder function **perform\_speaker\_diarization** for speaker diarization.
   * You can replace this with a suitable library/toolkit for actual speaker diarization.

**Challenges Faced:**

* Finding suitable datasets for emotion detection and speaker diarization.
* Implementing speaker diarization logic based on the available toolkit.

**Libraries/Tools Used:**

* Librosa for audio feature extraction.
* Scikit-learn for machine learning model.
* Pydub for audio file handling.
* Seaborn and Matplotlib for visualization.

**User Manual:**

1. **Uploading Audio Files:**
   * Ensure your audio files are in WAV format.
   * Replace the **audio\_files** list in the code with the paths to your uploaded audio files.
2. **Running the Code:**
   * Run the script and observe the Emotion Detection and Speaker Diarization results in the console output.

This prototype provides a foundation for your project, but you may need to extend it further based on your specific requirements, especially for speaker diarization, which requires a sophisticated approach.