Detailed Description

1. AudioProcessor: Handles loading and preprocessing of audio files.
   - load_audio(): Loads an audio file from the file path.
   - preprocess(): Preprocesses the loaded audio data (e.g., noise reduction, normalization).
2. FeatureExtractor: Extracts relevant features from audio data.
   - extract_mfcc(): Extracts Mel-frequency cepstral coefficients.
   - extract_chroma(): Extracts chroma features.
3. ModelTrainer: Manages training and tuning of machine learning models.
   - train_model(): Trains a machine learning model using the extracted features.
   - tune_model(): Tunes the model's hyperparameters to optimize performance.
4. Predictor: Uses the trained model to make predictions on new audio data.
   - predict(): Predicts the bird species from the audio data using the trained model.
5. WebApp: Manages the web application for user interaction.
   - run(): Runs the Flask web application.
   - upload_audio(): Handles audio file uploads from users.
   - get_results(): Returns prediction results to users.

These components work together to create a cohesive system for recognizing bird sounds, providing a user-friendly interface for uploading audio files and receiving accurate predictions. The architecture and design diagrams help in understanding the flow and interactions within the system, ensuring a well-organized and efficient implementation.