README

1 Setup and Running the Solution

To run this solution, follow the steps below:

1) **Download and Install Python**
2) **Install Required Libraries**
   - Ensure the following libraries are installed to handle audio-to-text transcription with Whisper and image-to-text extraction with EasyOCR.
   - Whisper: for automatic speech recognition (ASR).
   - EasyOCR: for optical character recognition (OCR) on images.
   - Open **cmd** from Windows and then you can install these libraries by running the following command:

```
pip install openai-whisper easyocr
```

3) **Install and Launch Jupyter Notebook**
   - Install Jupyter Notebook.
   - After installing Jupyter Notebook, open **cmd** from the location where Notebook is installed and run the following command to open it:

```
jupyter notebook
```

   - This will launch the Jupyter Notebook interface in your browser.
4) **Run the Solution**
   - Open the notebook file (Audio_Captcha.ipynb) from the Jupyter dashboard.
   - Set the dataset path according to your setup.
   - Run each block of codes sequentially to get the desired output.

2 High-Level Overview of the Approach

The methodology is broken down into the following steps:

1) **Finding Common Files**
   - Identify common filenames between audio and image datasets based on matching filenames (excluding extensions).
   - Save the list of common filenames to a CSV file for further processing.
2) **Audio Decoding with Whisper**
   - Preprocess audio files by normalizing volume and trimming silence.
   - Use the Whisper ASR model to transcribe the audio to text. Store the transcriptions in a CSV file for further analysis.
3) **Converting Decoded Audio Texts**
   - Clean and normalize the audio-transcribed texts by converting number words to digits and removing unwanted characters.
   - Save the cleaned transcriptions to a new CSV file for comparison.
4) **Image Text Extraction using OCR**
   - Preprocess images (convert to grayscale and resize) to prepare for text extraction.
   - Apply EasyOCR to extract text from images and save them in a CSV file.
5) **Converting Extracted Image Texts**
   - Clean the OCR-extracted texts by removing non-alphanumeric characters.
   - Save the cleaned image texts into a new CSV file for comparison with the audio transcriptions.
6) **Evaluation**
   - Merge audio-transcribed and image-extracted texts based on common filenames.
   - Calculate accuracy metrics such as Exact Match Accuracy, Levenshtein Distance, and Character Error Rate (CER) to compare the alignment of the two text sources.

3 Design Decisions

- **Whisper:** Selected for its high performance in automatic speech recognition across various languages and accents, offering robust transcription in noisy environments.
- **EasyOCR:** Chosen for its versatility in extracting text from images in multiple languages and formats, and its suitability for quick extraction tasks.
- **Comparison Metrics:** Besides accuracy, Levenshtein Distance and Character Error Rate were used to measure text similarity, accounting for minor discrepancies.