

SOUNDSCRIPT: AUDIO-TO-TEXT & SUMMARY GENERATOR

 SoundScript



Drag & drop your audio file here or
Browse Files
Selected: How are you.mpp3

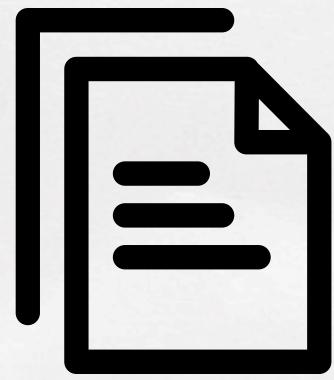
 Transcribed!  Summarized!

Transcription 

Hello, David. How are you today? I'm good, thank you. How are you? I'm fine, too, What are you up? Not much, just work. And you? Same. Busy with work. It was very nice talking you too. See you around, Sarah. Bye.

Summary 

David and Sarah exchanged greetings and briefly discussed their work. They concluded the conversation by saying goodbye and agreeing to see each other around.



PROBLEM STATEMENT

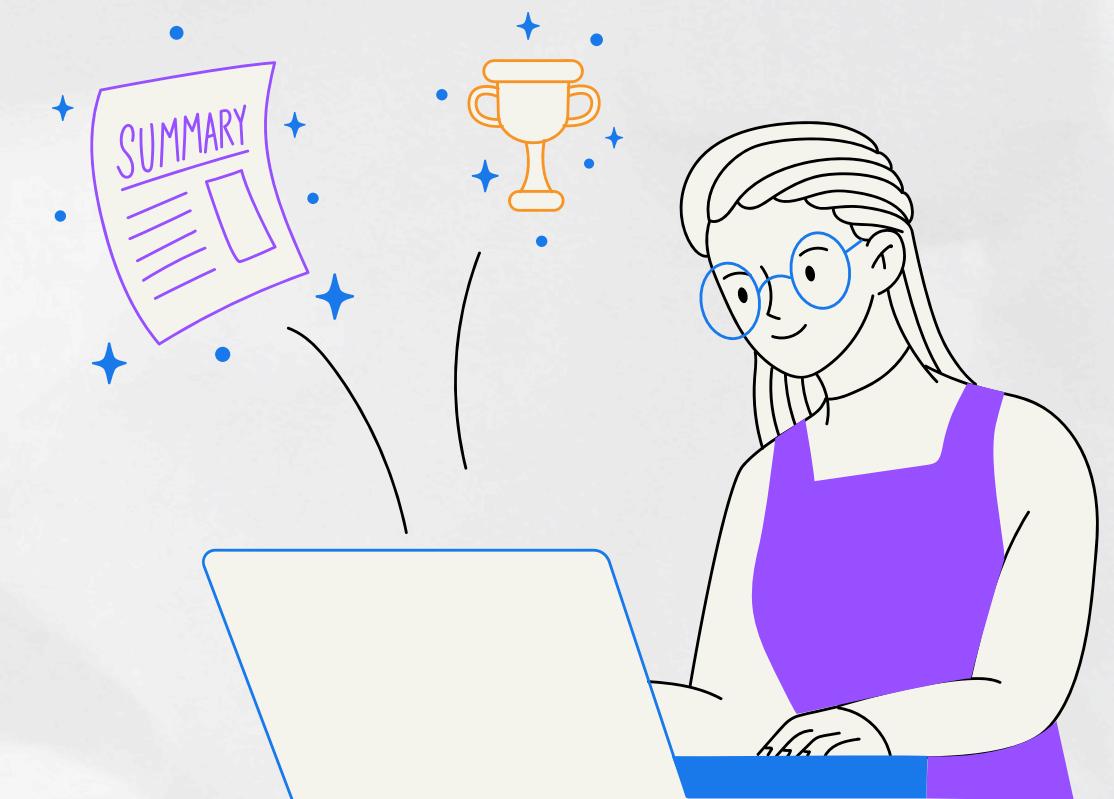
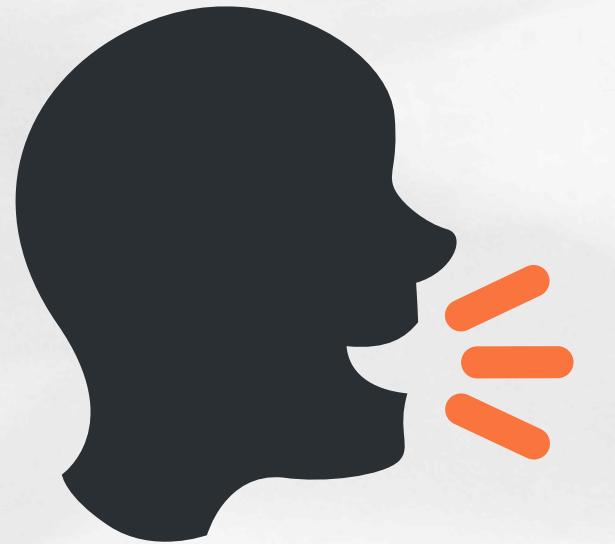


- “I wish I could **understand the language** that my teammates speak.”
- “I want to **review a long client call**, but re-listening to it is so **time-consuming**.”
- “I missed **critical details** in that recorded meeting because of the **background noise**.”
- “I sometimes **miss important nuances** in conversations when there's an **accent** I'm **not familiar with**.”
- “I need to share **key takeaways** from a voice recording, but typing it all out takes **forever**.”



KEY FEATURES

- High-Accuracy Audio Transcription
- Intelligent Conversation Summarization
- Secure & Private Handling
- Downloadable Outputs
- Intuitive & Interactive UI
- Clear Error Handling & Feedback



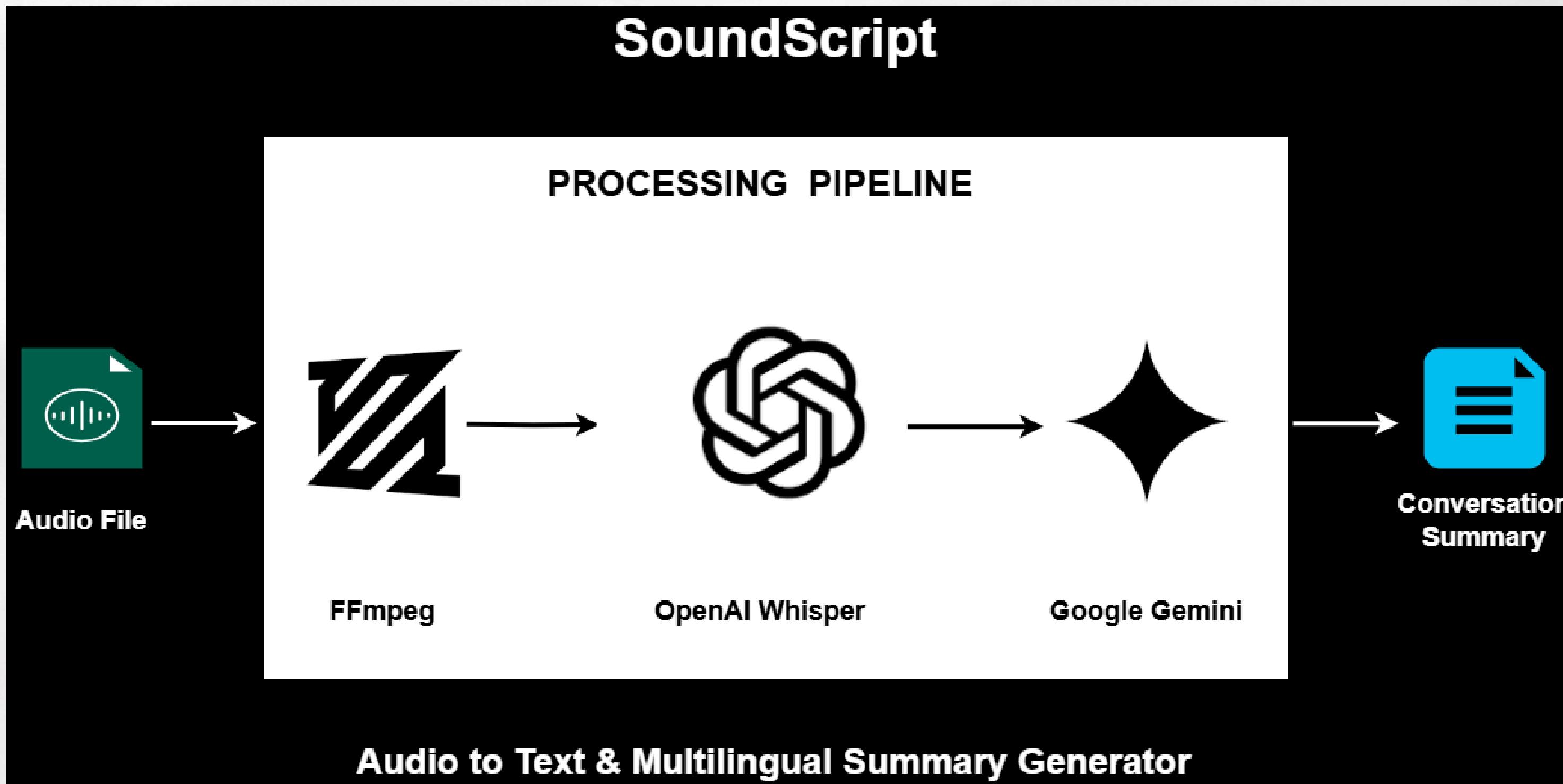
 info@edquest.co.in

  +91-9467752177

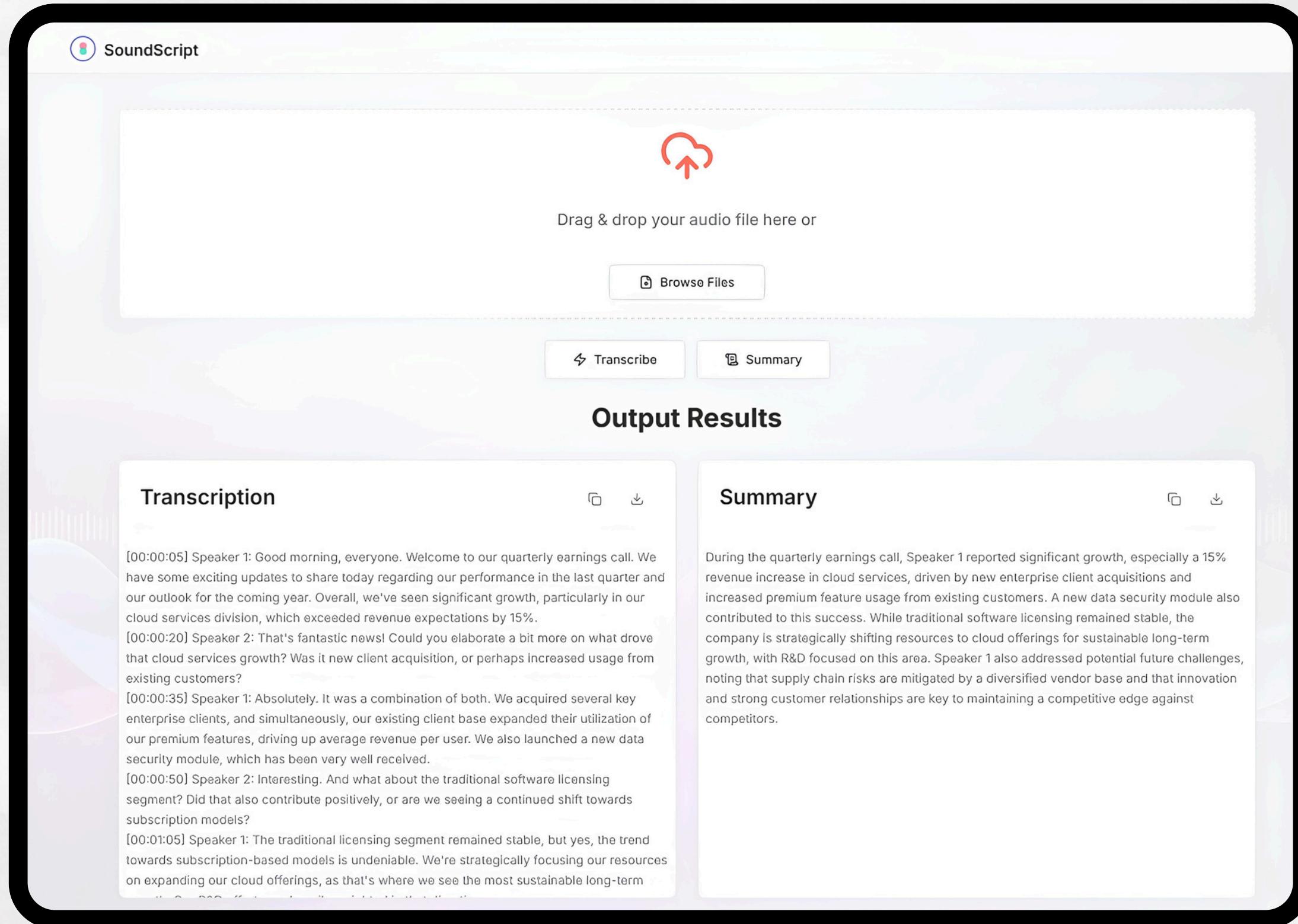
edquest

ARCHITECTURE

SoundScript



UI DESIGN



 info@edquest.co.in

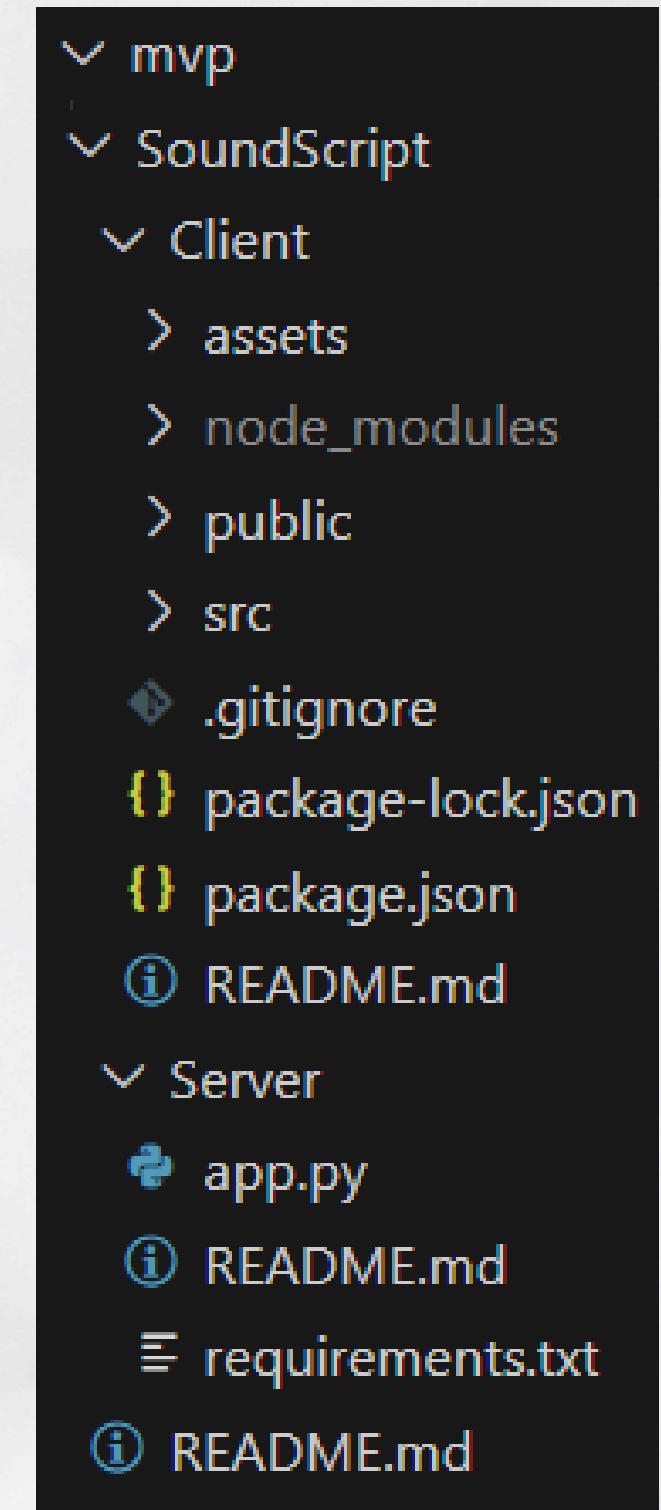
  +91-9467752177

edquest

PROJECT STRUCTURE

Key Files:

- **Server/app.py** - contains the server code
- **Client/src/app.js** - contains user functionalities code
- **Client/src/app.css** - contains the styling code
- **Server/requirements.txt** - to install all dependencies
- **Server/README.md** - How to run the server code
- **Client/README.md** - How to run the client code
- **README.md** - Overall review of the application



HOW TO RUN THE APPLICATION - SERVER

Steps :

- git clone <repo>
- cd Server
- python app.py

```
PS C:\Users\hi\Desktop\Whisper\mvp\SoundScript\Server> python app.py
Whisper model loaded successfully from local file.
Gemini API configured successfully.
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
Whisper model loaded successfully from local file.
Gemini API configured successfully.
* Debugger is active!
```

HOW TO RUN THE APPLICATION - CLIENT

Steps :

- open a new terminal
- cd client
- npm install
- npm start

```
PS C:\Users\hi\Desktop\Whisper\mvp\SoundScript\Client> npm start  
> frontend@0.1.0 start  
> react-scripts start
```

USAGE GUIDE

- Open the app
- Upload the audio file
- Click on “Transcribe” button
- Get Transcribed Text
- Click on “Summarize” button
- Get Summary of the audio
- Download Summary Text

The screenshot shows the SoundScript app interface. At the top, there's a logo with a microphone icon and the text "SoundScript". Below that is a large blue cloud icon with an upward arrow. To its right, there are two options: "Drag & drop your audio file here or" and "Browse Files". Underneath is a message "Selected: How are you.mpp3". At the bottom, there are two green buttons: "Transcribed!" with a checkmark and "Summarized!" with a checkmark. Below these buttons are two sections: "Transcription" and "Summary". The "Transcription" section contains a block of text: "Hello, David. How are you today? I'm good, thank you. How are you? I'm fine, too, What are you up? Not much, just work. And you? Same. Busy with work. It was very nice talking you too. See you around, Sarah. Bye." The "Summary" section contains a shorter text: "David and Sarah exchanged greetings and briefly discussed their work. They concluded the conversation by saying goodbye and agreeing to see each other around."

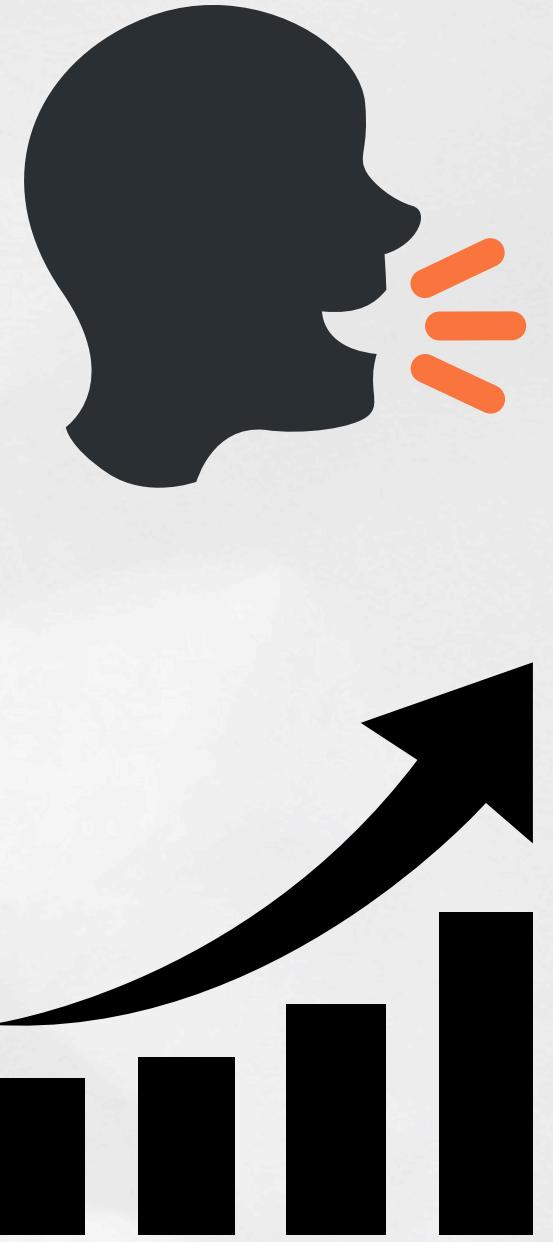
TECHNOLOGIES USED

- **React** – UI framework
- **Open AI Whisper** – Library for audio to text & vice versa
- **Python** – Programming language to write backend logic
- **FFMpeg** – Powerful audio processing tool
- **Google Gemini** – LLM model used



FUTURE SCOPES

- Social Media Tool for Translation
- Sentiment Analysis
- Keyword/Topic Extraction
- Advanced Audio Pre-processing
- Real-time Transcription & Summarization
- Mobile App for iOS/Android



THANK YOU

Contact Us



@edquestofficial



info@edquest.co.in



@edquestofficial



+91 - 9467752177



www.edquest.co.in