

# MeetAI Project Inputs

Feature/Component	Input	Source	Format	Details
<b>Real-Time Note-Taking (Smart Summarization)</b>				
- Audio Capture (WebRTC)	Raw audio stream	WebRTC via <code>getUserMedia</code> in React frontend	PCM audio in <code>MediaStream</code> , chunked as <code>Blob</code>	Captured via <code>MediaRecorder</code> , chunked every 30 seconds.
- Transcription (Whisper)	Audio chunks	Frontend sends <code>ArrayBuffer</code> to Node.js backend	Binary audio data (WAV)	Saved temporarily as a WAV file, processed by Whisper.
- Summarization (BART)	Text transcription	Whisper output in Node.js child process	Plain text string	Processed by BART to generate concise summaries.
- UI Display (React)	Summarized text	Node.js backend via Socket.io	JSON object	Rendered in real-time in the "Live Notes" panel.
<b>Video Conferencing Core</b>				
- Real-Time Communication	Audio and video streams	WebRTC via <code>getUserMedia</code>	<code>MediaStream</code> (audio/video)	Captured from participants' webcam and microphone.
- Meeting Rooms	User actions	Socket.io events from React frontend	JSON events	Events like joining or leaving a room.
<b>AI Teaching Assistant</b>				
- Real-Time Transcription & Translation	Audio streams	WebRTC via <code>getUserMedia</code>	PCM audio in <code>MediaStream</code> , WAV	Processed for transcription and optional translation.
- Automated Quiz Generation	Text transcriptions	Whisper output via backend	Plain text string	Used to generate quiz questions based on content.
- Student Engagement Analytics	Video frames	WebRTC video stream in frontend/backend	Video frame data (raw pixels)	Analyzed for facial features and engagement metrics.
<b>AI Coworking Tools</b>				
- Smart Summarization	Text transcriptions	Whisper output via backend	Plain text string	Processed to create meeting notes and action items.

Feature/Component	Input	Source	Format	Details
- Contextual Virtual Assistant	Chat messages or spoken queries	React frontend (chat input) or Whisper output	Plain text string	Processed to provide contextual responses.
- Live Code Collaboration	Code snippets	React collaborative editor (e.g., Monaco)	Plain text string	Used to generate AI-powered code suggestions.
<b>AI Moderation</b>				
- Detoxify & Timekeeping	Chat messages & meeting duration	React frontend chat input & server timestamp	Plain text & numeric timestamp	Analyzed for toxicity and used to track meeting time.

## Additional Notes

- **Primary Input:** Real-time audio and video streams from WebRTC.
- **Intermediate Inputs:** Audio chunks (WAV), text transcriptions, video frames.
- **Final Inputs:** JSON data (e.g., summaries, quizzes) to React UI.
- **Storage:** MongoDB for persistent transcriptions/summaries.
- **Processing:** Node.js with Python child processes for AI models.