© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716310

AI SaaS: An Automated Framework for Real-Time Video Conferencing and AI-Driven Meeting Summarization

HETAL AMRUTIA (PROJECT GUIDE)[1], VRATESH GHADGE[2], OM GHAG[3], ATHARVA KADAM[4]
[1,2,3,4]*Department of Computer Engineering, University of Mumbai, NHITM, Thane (W)*

Abstract- ***The AI SaaS Meeting Agent is an advanced video conferencing platform designed to enhance virtual collaboration through the integration of artificial intelligence. Similar to popular tools like Google Meet and Zoom, it allows users to create and manage video meetings effortlessly while offering intelligent post-meeting analytics. Once a meeting concludes, the system automatically processes the session to generate a comprehensive summary, detailed transcript, and accessible recording of the discussion. These features eliminate the need for manual note-taking and ensure that all essential points, ideas, and decisions are accurately documented and easy to review later.The platform's cloud-based SaaS architecture provides scalability, flexibility, and accessibility, making it suitable for organizations, educators, and teams working across diverse domains. A key feature of the platform is its built-in AI chatbot assistant, which serves as a virtual helper for users throughout their meeting experience. The chatbot assists with common tasks such as scheduling, navigation, troubleshooting, and answering queries related to meeting management. It leverages natural language processing (NLP) to interact intelligently with users, providing real-time guidance and support.***

I. INTRODUCTION

The AI SaaS Meeting Agent is an advanced video conferencing platform designed to enhance virtual collaboration through the integration of artificial intelligence.

Similar to popular tools like Google Meet and Zoom, it allows users to create and manage video meetings effortlessly while offering intelligent post-meeting analytics.

Once a meeting concludes, the system automatically processes the session to generate a comprehensive summary, detailed transcript, and accessible recording of the discussion.

These features eliminate the need for manual note-taking and ensure that all essential points, ideas, and decisions are accurately documented and easy to review later.

The platform's cloud-based SaaS architecture provides scalability, flexibility, and accessibility, making it suitable for organizations, educators, and teams working across diverse domains.

A key feature of the platform is its built-in AI chatbot assistant, which serves as a virtual helper for users throughout their meeting experience.

The chatbot assists with common tasks such as scheduling, navigation, troubleshooting, and answering queries related to meeting management.

1.1.2 Objectives

The goal of the AI SaaS Agent is to build a smart, cloud-based system that can automate everyday workflows and make smarter decisions with the help of AI/ML. It will support multiple users at once, ensure secure access, and offer subscription-based services with round-the-clock availability. Ultimately, the agent aims to improve efficiency, cut down on manual effort, and provide a seamless, reliable experience for users.

Specific objectives include:

1. Develop Real-Time AI Interaction: To enable users to create and deploy custom AI agents that can join video calls as active participants, offering real-time assistance, coaching, and interaction.
2. Automate Post-Meeting Workflow: To automatically generate and provide users with a complete meeting package, including a

searchable transcript, an AI-powered summary, and a video recording immediately after the call concludes.

3. Create a Unified User Experience: To build a single, cohesive platform that combines video calling, AI agent management, and post-meeting analysis, eliminating the need for separate third-party tools.
4. Implement Conversational Intelligence: To allow users to interact with a chat-based AI that can answer questions and provide insights based on the content of past meetings.
5. Build a Secure & Scalable Platform: To implement robust user authentication, a secure payment and subscription model, and ensure the application is scalable using a modern, full-stack technology architecture.

1.2 Proposed System

1.2.1 System Framework:

1. Unified User Dashboard: Upon secure authentication (managed by Better Auth), users will land on a central dashboard. This will be their command centre for scheduling new meetings, creating and managing their custom AI agents, and accessing the repository of past meetings, including recordings, transcripts, and summaries.
2. AI Agent Creation Studio: A core innovation of the platform is an intuitive interface where users can design and configure their AI agents. Users can define an agent's name, persona (e.g., "Socratic Tutor, " "Agile Coach, " "Executive Assistant"), and specific instructions or knowledge bases to guide its behaviour during calls.
3. Real-Time Conferencing Engine: Leveraging the Stream SDK, the platform will provide high-quality, low-latency video and audio calling. The crucial distinction is its native integration with the AI engine, allowing the created AI agents to join calls as distinct participants.
4. Live Interaction and Intelligence Layer: During a live call, the OpenAI API will power the AI agent's ability to listen to the conversation, process information in real-time, and respond vocally or textually based on its predefined persona. This enables active participation, such as asking clarifying questions, providing data-driven insights, or guiding the discussion.
5. Automated Post-Meeting Intelligence Hub: Once a meeting concludes, background jobs managed by Inngest will automatically process the session data. This hub will provide users with: A full, searchable transcript of the conversation. An AI-generated summary highlighting key topics, decisions, and action items. A full video recording of the meeting. A conversational AI chat feature, allowing users to ask questions about the content of their past meetings (e.g., "What did we decide about the Q4 budget in last week's call?").
6. Subscription & Monetization: The platform will be built as a commercial SaaS product, using Polar to manage secure subscription tiers and payments, allowing access to premium features like an increased number of AI agents, longer meeting durations, or advanced analytics.
7. By integrating these components, Meet AI will provide a holistic and intelligent meeting experience. It moves beyond passive recording to offer active, personalized AI collaboration, transforming how individuals and teams learn, strategize, and work together.

1.2.2 Tools and Technologies

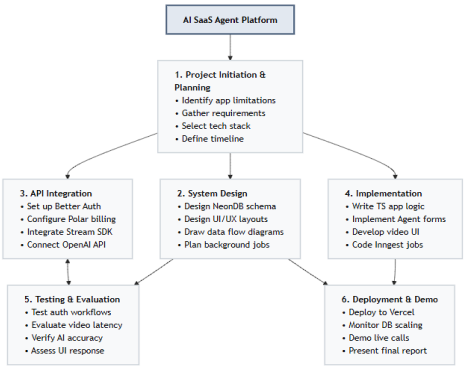
Table 1.1 - Tools and Technologies

Layer	Technology
Frontend & Backend Framework	Next.js 15 (TypeScript)
Database Storage	PostgreSQL hosted via NeonDB
ORM (Object-	Drizzle ORM

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716310

Layer	Technology
Relational Mapping)	
Live Video Conferencing Service	Stream SDK
Artificial Intelligence Models	OpenAI API
Authentication Management	Better Auth
Billing & Subscriptions	Polar
Background Job Processing	Inngest
Hosting & Deployment	Vercel

1.2.3 Methodology



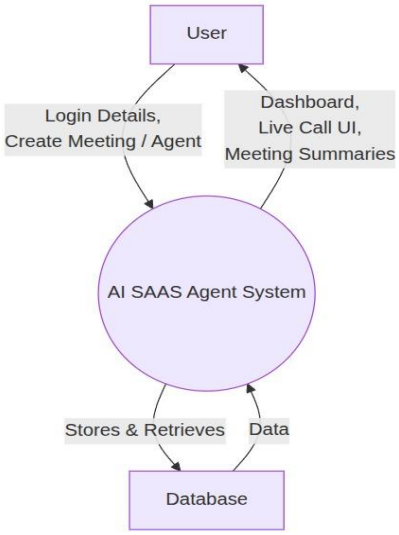
The methodology for developing the AI SaaS platform followed a phased, agile lifecycle focused on building a secure foundation before integrating complex real-time video and AI modules. The development lifecycle was structured into six key phases:

a) Project Initiation & Planning: Identified the limitations of passive video conferencing tools and established the core requirements and tech stack (Next.js, NeonDB, Stream SDK, OpenAI).
b) System Design: Designed the database schema using Drizzle ORM and mapped out the data flow pipelines for the video and AI integrations.
c) API Integration: Configured secure user authentication via Better Auth and integrated the core SaaS APIs, including Stream SDK for video and Polar for billing.
d) Implementation: Developed the frontend UI and backend TypeScript logic, including custom AI Agent creation forms and background jobs via Inngest.
e) Testing & Evaluation: Evaluated the latency of Stream video calls, verified the accuracy of AI-generated summaries, and tested database read/write speeds
f) Deployment: Deployed the final Next.js application to Vercel's edge servers and monitored the live production database on NeonDB.

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716310

1.2.3 Design Details

Level-0 DFD



Level-1 DFD

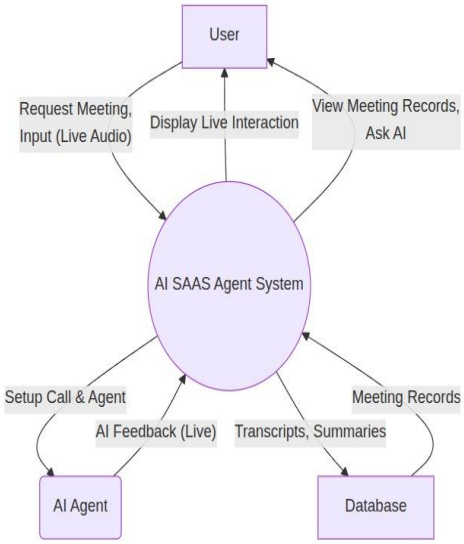


Figure 1.2 – System Design Workflow

The system design of the AI SaaS Agent platform is modeled using Data Flow Diagrams (DFDs) to illustrate how information moves between the user, the core system, the AI modules, and the database.

1. High-Level Architecture (Level-0 DFD Workflow)

The Level-0 DFD represents the context diagram, providing a bird's-eye view of the entire system's interactions with external entities.

- Input & Configuration: The workflow begins with the User providing essential inputs, specifically "Login Details" for authentication and commands to "Create Meeting / Agent" to the central AI SaaS Agent System.
- System Output: In response, the core system processes these inputs and serves the graphical interface back to the User, which includes the "Dashboard," the "Live Call UI," and post-call "Meeting Summaries."
- Data Persistence: Concurrently, the central system communicates with the Database entity, functioning as the persistent storage layer. It continuously "Stores & Retrieves Data" to ensure user profiles, agent configurations, and session states are maintained securely.

2. Detailed Operational Process (Level-1 DFD Workflow)

The Level-1 DFD breaks down the central AI SaaS Agent System into a more granular, step-by-step workflow, highlighting the live interaction loop and post-meeting data routing. This can be categorized into three distinct phases:

- Phase 1: Live Call Initiation & Orchestration
  - The User initiates the process by sending a "Request Meeting" to the system.
  - The system immediately triggers the AI Agent entity by sending a "Setup Call & Agent" payload, configuring the AI persona for that specific session.

- Phase 2: Real-Time Interaction Loop
  - During the live session, the User provides continuous "Input (Live Audio)" into the system.
  - The system routes this data to the AI Agent, which processes the context and returns "AI Feedback (Live)."
  - The system synthesizes the user's audio/video and the AI's feedback to "Display Live Interaction" back to the User seamlessly.

- Phase 3: Post-Meeting Data Processing & Retrieval

- Upon the conclusion of the meeting, the core system generates the final text data and pushes the "Transcripts, Summaries" into the Database for permanent storage.
- When the User wishes to review past sessions, the system fetches the archived "Meeting Records" from the Database.
- Finally, the system outputs these records, allowing the User to "View Meeting Records" and utilize the interactive "Ask AI" feature based on the historical context.

1.2.5 Hardware and Software Setup

Hardware Requirements:

- Modern Laptop or Desktop computer.
- Multi-core Processor.
- Minimum of 8 GB RAM.
- Stable high-speed Internet Connection.

Software Requirements:

- Operating System: Windows 10/11, macOS, or Linux.
- Runtime Environment: Node.js v22.15.1 (via nvm recommended) along with bundled npm/npx.
- Development Tools: Git for version control, a Code Editor , a Terminal/CLI, and a modern Web Browser.
- Networking Tools: ngrok for local webhook testing and routing

II. RESULT AND DISCUSSION

2.1.1 Implementation Plan

The execution of the project was divided into seven distinct phases to ensure systematic integration of features:

Phase 1: Core foundation and auth setup implemented with complete sign-in/sign-up UI

Phase 2: Finalized social authentication connections for Google and GitHub.

Phase 3: Designed and developed the responsive Dashboard Sidebar and Navbar for user navigation.

Phase 4: Configured the tRPC backend communication and built the Agent setup forms and dialog components.

Phase 5: Implemented Agent update/delete features, filters, and integrated the Meeting creation forms.

Phase 6: Developed Meeting Data Tables and integrated the Stream SDK for core video call functionality

Phase 7: Connected the custom AI Agents with the Stream video feed and OpenAI integrations, while configuring Inngest for asynchronous background jobs.

2.2 Result

The developed AI SaaS Agent platform successfully transformed traditional video conferencing by seamlessly integrating real-time communication with active artificial intelligence. The Stream SDK accurately handled low-latency video and audio routing, while the OpenAI integration effectively empowered custom AI personas to participate and respond during live calls. The serverless Next.js architecture enabled a responsive user dashboard, securely managed through Better Auth. System testing demonstrated reliable background processing via Inngest, ensuring quick and accurate generation of post-meeting transcripts and automated summaries. The interactive Ask AI feature helped users rapidly retrieve critical meeting insights, improving knowledge retention. Overall, the system proved to be a practical and highly efficient SaaS solution for enhancing remote collaboration and overall team productivity.

2.2.1 Outputs (Website)

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716310

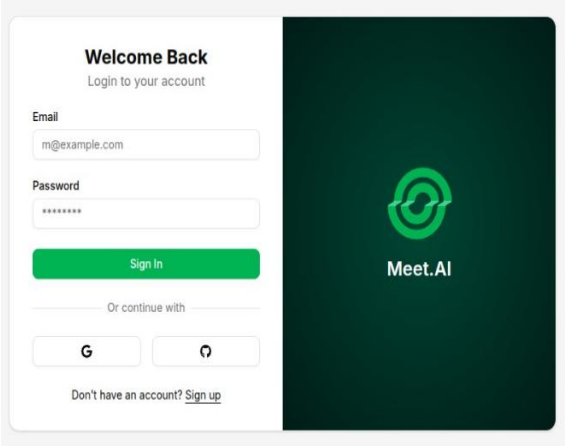


Figure. 2.1 – Sign-in Page

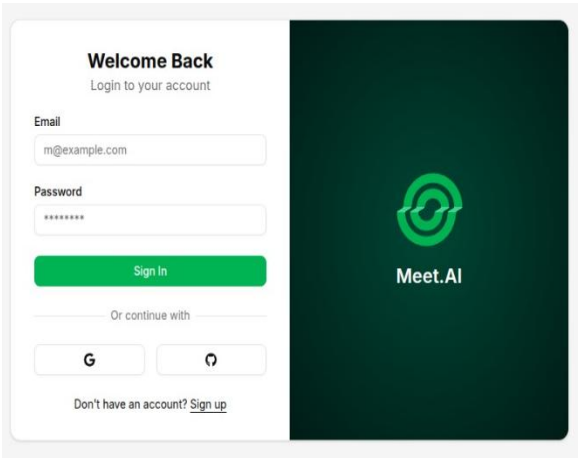


Figure. 2.2 – Sign-up Page

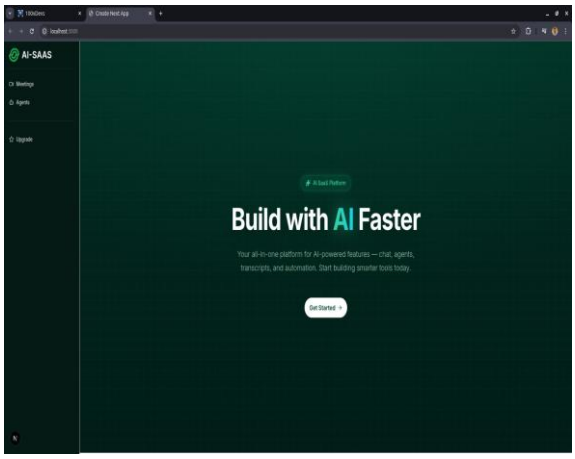


Figure. 2.3 – Home Page

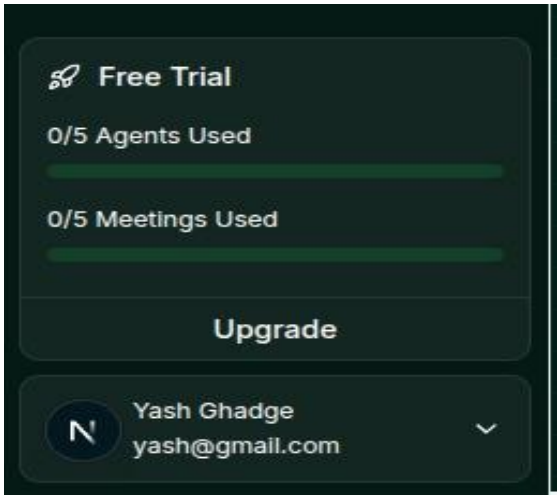


Figure. 2.4 – Dashboard

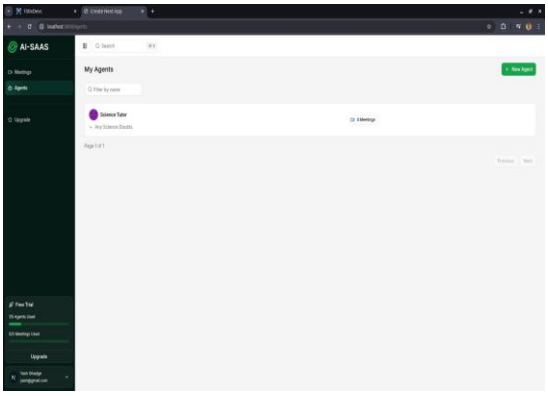


Figure. 2.5 – Agents Page

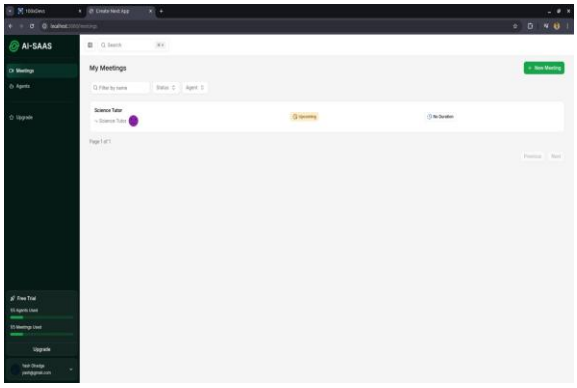


Figure. 2.6 – Meetings Page

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716310

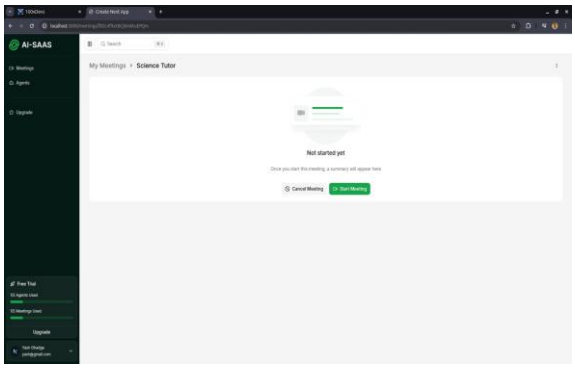


Figure. 2.7 – Meetings Scheduling Page

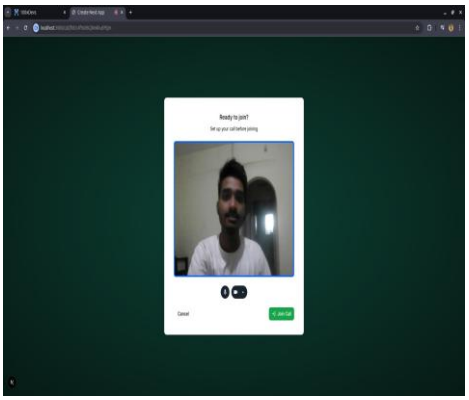


Figure. 2.8 – Meeting Joining Page

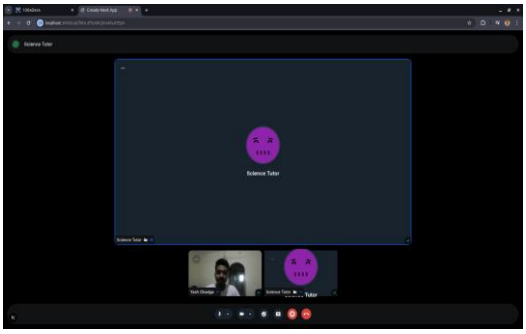


Figure. 2.9 – Meeting Page

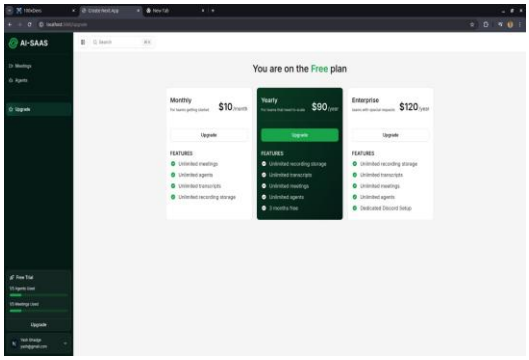


Figure. 2.10 – Upgrade Plans Page

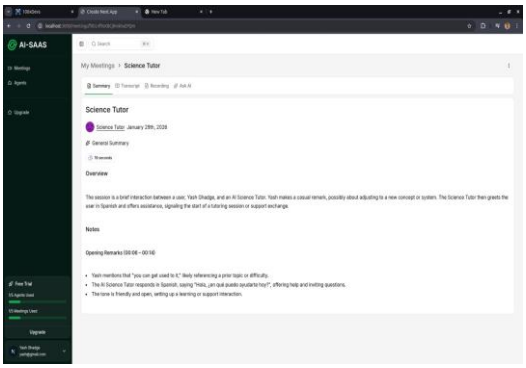


Figure. 2.11 – Summary Page

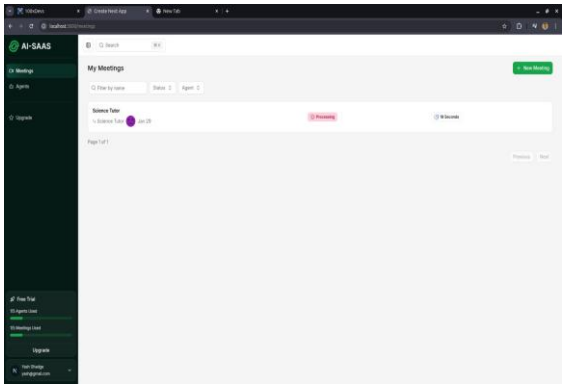


Figure. 2.12 – Meetings Page (After Completion)

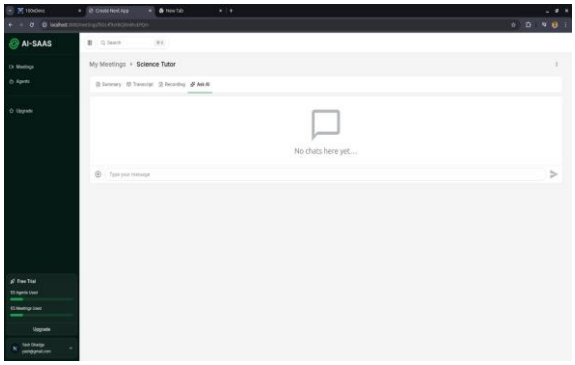


Figure. 2.13 – ASK AI Page

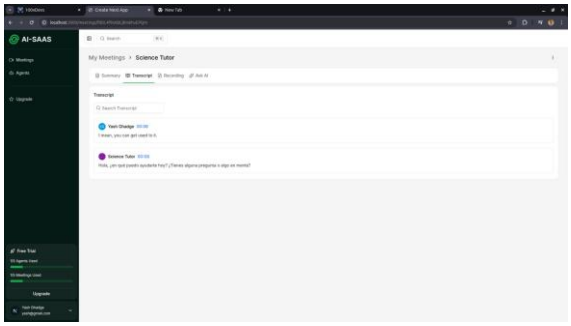


Figure. 2.15 – Transcript Page

CONCLUSION

The developed AI SaaS platform effectively addresses the limitations of conventional video conferencing tools by transitioning Artificial Intelligence from a passive, post-meeting utility into an active, real-time participant. By leveraging a modern, serverless Next.js architecture integrated with the Stream SDK and OpenAI, the system enables users to deploy customizable AI personas directly into live calls for immediate assistance. Furthermore, the automated post-meeting workflow seamlessly generates actionable summaries, accurate transcripts, and video recordings, significantly reducing the cognitive load of manual note-taking. This unified approach to communication and intelligent analysis successfully demonstrates a scalable, highly efficient framework for the future of remote collaboration and productivity.

REFERENCES

[1] M. K. Lee, S. A. O'Keefe, and R. L. Mandryk, "Observe, Ask, Intervene: Designing AI Agents for More Inclusive Meetings," arXiv preprint arXiv:2501.10553, Jan. 2025.

[2] Y. Zhang, R. Takanobu, and M. Huang, "Summarizing Speech: A Comprehensive Survey," arXiv preprint arXiv:2504.08024, Apr. 2025.

[3] H. Chen, Z. Wang, and Y. Li, "Online Meeting Summarization with Large Language Models," arXiv preprint arXiv:2502.03111, Feb. 2025.

[4] A. R. Fabbri, S. Li, and D. Radev, "Benchmarking Large Language Models for Meeting Assistance," arXiv preprint arXiv:2502.04376, Feb. 2025.

[5] Guokan Shang et al., "Unsupervised Abstractive Meeting Summarization with Multi-Sentence Compression …," arXiv preprint, May 2018.

[6] J. Xie, G. Sun, and J. Han, "Abstractive Meeting Summarization Using Neural Networks," IEEE Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2729–2740, 2020

[7] K. Krishna, G. Ganesan, and M. S. Ayyar, "Automatic Meeting Minutes Generation Using Natural Language Processing," International Journal of Research in Engineering and Technology (IJRET), vol. 9, no. 4, pp. 115–120, Apr. 2020.

[8] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 38–45, 2020.

[9] Y. Liu and M. Lapata, "Text Summarization with Pretrained Encoders," Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3721–3731, 2019.

[10] R. Carenini, G. Murray, and R. Ng, "Methods for Mining and Summarizing Spoken Meetings," IEEE Data Engineering Bulletin, vol. 34, no. 3, pp. 3–13, 2011.