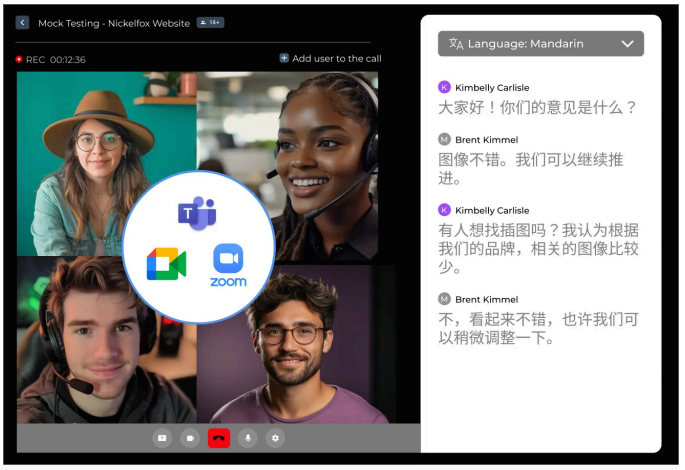
Case Study

Generic Meeting Bot for Captions Capture and Reporting



SpringCT partnered with a leading technology company to develop a versatile meeting bot capable of joining Microsoft Teams, Zoom, and Google Meet meetings. The primary objective was to enable seamless meeting participation by the bot as a user, capture meeting captions, and report the data to a backend server for advanced AI-based processing. This innovative solution required no changes to tenant settings, ensuring smooth adoption and integration. The first phase focused on Microsoft Teams, with provisions for later support of Zoom and Google Meet with minimal modifications.

Product Features

The bot application was designed with a robust set of features to meet the client's requirements:

- **Multi-Platform Compatibility**
  Initially developed for Microsoft Teams, with a scalable architecture enabling quick support for Zoom and Google Meet.
- **Headless Browser Agent**
  Utilized Puppeteer to enable the bot to join meetings as a user and interact with meeting interfaces.
- **Real-Time Caption Capture**
  Captures meeting captions in real-time for transcription and reporting.
- **Effortless Deployment**
  Requires no configuration changes to tenant settings, ensuring easy deployment and adoption.
- **Azure Cloud Integration**
  Fully deployed on Azure Cloud, leveraging scalable and reliable cloud infrastructure.

Key Technical Achievements

- **Headless Browser Implementation**
  Configuring Puppeteer to reliably join meetings across different platforms and handle dynamic meeting interfaces without human intervention.
- **Platform-Specific Differences**
  Addressing UI and API differences between Teams, Zoom, and Google Meet while maintaining a consistent backend integration.
- **Real-Time Caption Capture**
  Ensuring accurate and timely capture of captions from the meeting interface and streaming it to the backend server.
- **Azure Event Integration**
  Implementing Azure Event Grid and Event Bus for efficient communication and processing of captured data in real-time.
- **Scalability and Deployment**
  Orchestrating services using Azure Kubernetes Service (AKS) to support high availability and scalability.

Technologies Used

- **Frontend:** Headless browser agent implemented using Puppeteer.
- **Backend:** Node.js with integration to Azure Event Grid and Azure Event Bus.
- **Cloud Infrastructure:** Azure Kubernetes Service (AKS) for container orchestration, Azure App Service for deployment, and Cosmos DB for data

storage.

Results

The generic meeting bot delivered the following measurable outcomes

- **Enhanced Automation: The bot's ability to join meetings autonomously and capture captions significantly reduced manual effort.**
- **Cross-Platform Scalability: The architecture supported seamless extension to Zoom and Google Meet with minimal development effort.**
- **Operational Efficiency: Deployment in Azure Cloud ensured high reliability, scalability, and ease of management.**
- **AI-Driven Insights: Real-time caption data fed into the backend server provided valuable inputs for advanced AI-based analytics and processing.**

Conclusion

SpringCT's generic meeting bot application showcases its ability to address complex challenges in automating meeting participation and caption capture across multiple platforms. By leveraging headless browser technology, a scalable Azure-based architecture, and innovative design, the solution enabled the client to streamline operations and unlock valuable insights. The successful implementation in Microsoft Teams and planned extension to Zoom and Google Meet reflect the bot's adaptability and efficiency, setting the stage for future advancements in meeting automation and AI-driven communication solutions.