​**Skynote**

Introduction

Modern enterprises and teams hold frequent meetings, and manually recording meeting minutes is not only time-consuming and inefficient but also prone to missing key information. This project aims to develop an intelligent meeting assistant that integrates AI technology into the Skype for Business platform. By leveraging speech recognition and natural language processing (NLP), the system will enable automatic transcription and summarization of meeting content. The project objectives include reducing the burden of manual note-taking, improving meeting efficiency, and ensuring that participants can easily access clear and accurate meeting records and key points afterward.

With the rise of remote and hybrid work models, more organizations require AI-powered tools to support virtual meeting collaboration. Market research indicates that the AI meeting assistant sector is experiencing rapid growth, with a market size of approximately $2.5 billion in 2023, projected to reach $12.7 billion by 2031. Therefore, this project holds significant practical relevance and promising application potential.

Market Research

Based on the three core feature requirements—real-time speech-to-text transcription, AI-driven meeting summarization, and summarization of individual opinions—current meeting assistant software can be categorized into the following three types:​

**I. Direct Integration Solutions: Built-in Features in Real-Time Video Conferencing Platforms**

These platforms integrate the core features directly into their video conferencing services, allowing users to utilize them without installing additional plugins or third-party tools.​

**1. Zoom**

*•Real-time Speech-to-Text Transcription*: Provides live transcription during meetings, converting spoken words into text in real-time.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting summaries post-session and emails them to the host.​

*• Summarization of Individual Opinions*: Identifies key discussion points, important conclusions, and action items through its smart recording feature.​

**2. Google Meet**

*• Real-time Speech-to-Text Transcription*: Offers live captioning using AI technology to provide real-time subtitles during meetings.​

*• AI-Driven Meeting Summarization*: Features like "Take notes for me" automatically capture meeting highlights and organize them into Google Docs.​

*• Summarization of Individual Opinions*: Provides a "Summary so far" feature to help latecomers quickly catch up on discussed content.​

**3. Microsoft Teams**

*• Real-time Speech-to-Text Transcription*: Offers live transcription, converting spoken content into text during meetings.​

*• AI-Driven Meeting Summarization*: Through its intelligent recap feature, it automatically generates meeting summaries, identifies key topics, and creates chapters.​

*• Summarization of Individual Opinions*: AI-generated notes capture discussion points and action items.​

**4. Tencent Meeting**

*• Real-time Speech-to-Text Transcription*: Supports real-time speech-to-text conversion, providing live subtitles.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting minutes post-session, extracting key content.​

*• Summarization of Individual Opinions*: Intelligently analyzes speakers' viewpoints and opinions.​

**5. Feishu Meeting**

*• Real-time Speech-to-Text Transcription*: Provides live transcription, recording meeting content in real-time.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting minutes, extracting key topics.​

*• Summarization of Individual Opinions*: Capable of distinguishing different speakers and organizing their respective viewpoints.​

**II. Indirect Integration Solutions: Implemented Through Plugins or Third-Party Tools**

These tools integrate with mainstream video conferencing platforms, acting as plugins or third-party services to provide the core features.​

**1. Otter.ai**

*• Real-time Speech-to-Text Transcription*: Can automatically join Zoom, Google Meet, or Microsoft Teams meetings to transcribe conversations in real-time.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting summaries, capturing key points.​

*• Summarization of Individual Opinions*: Highlights key points and allows tagging teammates to assign action items.​

**2. Fireflies.ai**

*• Real-time Speech-to-Text Transcription*: Automatically records and transcribes meetings from various video conferencing applications.​

*• AI-Driven Meeting Summarization*: Generates meeting summaries, capturing key discussion points.​

*• Summarization of Individual Opinions*: Supports collaboration features, allowing users to add comments, tags, and emojis to specific parts of the conversation.​

**3. Fathom**

*• Real-time Speech-to-Text Transcription*: Records and transcribes meetings on Zoom, Google Meet, and Microsoft Teams.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting summaries, extracting key content.​

*• Summarization of Individual Opinions*: Identifies and highlights key points and action items during meetings.​

**4. Tactiq**

*• Real-time Speech-to-Text Transcription*: Provides live transcription for Google Meet, Zoom, and Microsoft Teams.​

*• AI-Driven Meeting Summarization*: Generates summaries with one click and creates follow-up emails containing action items.​

*• Summarization of Individual Opinions*: Utilizes AI to extract insights, capturing and tagging each participant's contributions.​

**III. Post-Processing Solutions: Analysis Through Uploaded Audio or Video Files**

These tools do not participate directly in real-time meetings but transcribe and analyze content through user-uploaded audio or video files.​

**1. Alibaba Tongyi Tingwu**

*•Real-time Speech-to-Text Transcription*: Supports multi-language transcription and translation, providing real-time subtitles and intertranslation in languages like Chinese and English.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting summaries, extracting key content.​

*• Summarization of Individual Opinions*: Capable of distinguishing different speakers and organizing their respective viewpoints.​

**2. Trint**

*• Real-time Speech-to-Text Transcription*: Transcribes video, audio, or live speech into over 40 languages.​

*• AI-Driven Meeting Summarization*: Automatically generates meeting summaries, capturing key points.​

*• Summarization of Individual Opinions*: Supports editing, collaboration, and integration with real-time workflows.​

Choosing the appropriate meeting assistant software should be based on your specific needs and existing workflows. If you aim to implement these features directly within real-time video conferences, opting for direct integration solutions may be more convenient. Conversely, if you seek in-depth analysis of meeting content post-session, post-processing solutions might be more suitable.

Market Demand and Industry Trends

Regardless of the solution, the demand for automated meeting transcription and intelligent summarization is growing rapidly. On one hand, enterprises seek to improve meeting efficiency by freeing employees from the burden of manual note-taking, allowing them to focus on discussions and decision-making. On the other hand, with the rise of remote and hybrid work models, meeting minutes have become a crucial component of both synchronous and asynchronous collaboration. Automated tools ensure that absent participants can stay informed about key discussions in a timely manner.

Research indicates that AI meeting assistants are evolving beyond simple transcription to provide deeper insights, including key point summarization, decision and action item extraction, and even task tracking based on meeting content. This trend is reflected in the functionalities of industry-leading products. For example, **Otter.ai** supports real-time transcription, multilingual recognition, and automatic summarization. **Tencent Meeting's AI Assistant Pro** generates real-time summaries during meetings, highlights participants' key concerns, and compiles action items post-meeting.

Overall, the industry is moving towards the deep integration of AI meeting assistant functionalities within major collaboration platforms to meet the increasing market demand.

Implementation Goals

Implementation Goals

Based on the aforementioned research, this project aims to develop a highly efficient and practical intelligent meeting transcription and summarization system with the following core functionalities:

**1. Real-time Speech-to-Text Transcription**  
Utilizing speech recognition technology, the system will convert spoken words into text in real time during meetings. It must support multi-speaker recognition and speaker diarization to ensure that every participant's speech is accurately recorded. The transcribed text will be displayed in the user interface, allowing attendees to follow along, make corrections if necessary, and highlight key points.

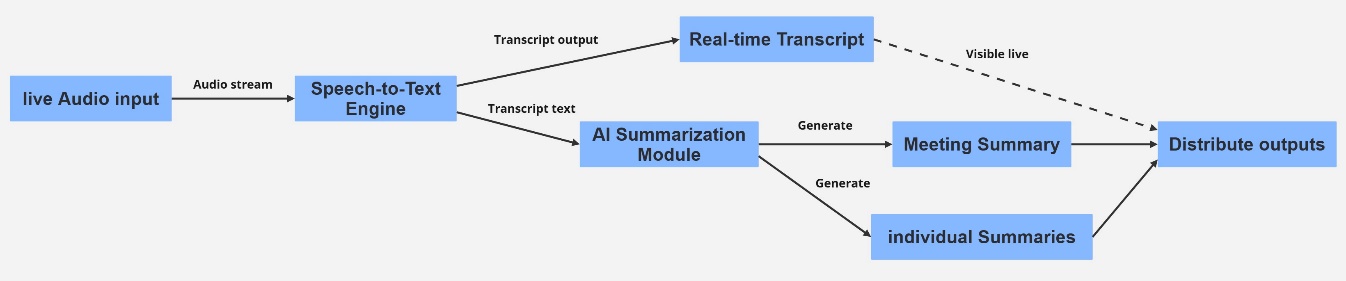
**2. AI-Powered Meeting Summarization**  
At the end of each meeting, the system will automatically generate a structured meeting summary based on the full transcription. The summary should extract key topics, decisions, conclusions, and important action items discussed during the meeting. By leveraging advanced natural language processing (NLP) and generative AI models, the summarization should be concise yet accurately reflect the discussion. For longer meetings, the system may offer sectional summaries or agenda-based summaries to ensure clarity and readability.

**3. Summarization of Individual Opinions**The system should generate an overview of each participant's contributions to the meeting. This feature will extract key opinions, concerns, or assigned tasks for each speaker and present them as brief summaries. This functionality is particularly valuable for decision-making meetings, as it clearly presents the viewpoints of different stakeholders. To achieve this, the system will classify transcription data by speaker and apply NLP techniques to distill each participant’s key statements.

**4. Action Item and Task Tracking (Optional)**  
To enhance the execution of meeting outcomes, the system can automatically detect action items discussed in the meeting (such as task assignments and deadlines) and generate a to-do list. Each task should specify the responsible person and due date, making it easier to export into team task management tools. While this feature is of lower priority in the initial implementation, it will be considered if time allows, providing a more comprehensive meeting solution.

These core features will work together to create an end-to-end AI-powered meeting assistant, ensuring that participants receive AI support throughout the entire meeting lifecycle—from real-time transcription to post-meeting summaries and action tracking.

Workflow Diagram



As shown in the diagram, the workflow of this system consists of the following stages:

**1. Audio Capture and Input:**  
At the start of a meeting, the system connects to the audio source (e.g., capturing live sound through a microphone or acquiring raw audio streams from a video conferencing platform’s audio interface). The output of this stage is a real-time audio data stream.

**2. Speech-to-Text Conversion:**  
The audio stream is transmitted in real-time to the speech recognition engine. The engine, based on a pre-trained Automatic Speech Recognition model (supporting multiple languages such as Chinese and English), converts speech into text. Additionally, the recognition process should differentiate between speakers and attach speaker labels to the transcribed text. This stage continues throughout the entire meeting, continuously generating real-time text transcriptions. Participants can view the word-by-word transcription on the front-end interface for real-time discussion tracking (as indicated by the dashed lines in the diagram).

**3. Data Storage and Processing:**  
The real-time transcribed text is temporarily stored and saved in the system backend for subsequent summarization. Text fragments may be cached during the meeting and processed when the meeting concludes or at designated trigger points. In case of network instability, the system should support local caching to prevent data loss.

**4. AI-Powered Summarization:**  
As the meeting concludes or reaches its final stages, the system triggers the summarization module. This module utilizes pre-trained Natural Language Processing (NLP) models (which can be fine-tuned Transformer-based models) to analyze and extract key information from the full transcript.

**5. Output and Distribution:**Once the summaries are generated, the system organizes and delivers the results. The outputs include:

•Displaying key meeting points and speaker overviews on the Web/App interface.

•Providing full-text transcriptions and summaries for download (e.g., generating PDF/Markdown meeting minutes).

•One-click sharing, such as sending the meeting summary via email to all participants or synchronizing it with collaboration platforms (e.g., Slack, DingTalk).

•If action items are detected, the system can generate a task list and push it to project management tools.

At this stage, the workflow is complete, achieving fully automated meeting transcription and summarization from speech input to structured meeting notes.

Task Time Estimation

To ensure an orderly development process, the overall project has been broken down into several key tasks, with estimated time requirements as follows:

**1. Requirement Analysis & Technology Selection (1 week)**  
This phase involves finalizing functional requirements, selecting appropriate speech recognition and AI summarization models, and designing the system architecture. Key considerations include ASR engine selection (e.g., Whisper, Google Speech API) and choosing NLP models for summarization (extractive vs. generative, fine-tuning needs).

**2. Core System Development (6 weeks)**

**• Real-Time Speech-to-Text Module (2 weeks):** Implement audio capture, real-time transcription, and speaker diarization. Focus on accuracy, latency reduction, and noise handling.

**• Backend & Database (1 week):** Develop a scalable backend architecture and storage model to handle real-time transcripts and meeting data.

**• Frontend UI Development (1 week):** Create an interactive interface to display real-time transcriptions, speaker identification, and keyword highlights.

**•AI Summarization & Opinion Extraction (2 weeks):** Implement automated meeting summarization using pre-trained NLP models (BERT, GPT, or TextRank), ensuring clear extraction of key discussions, decisions, and action items.

**3. System Integration & Optimization (2 weeks)**

**•** Ensure seamless interaction between real-time transcription, backend processing, and AI-generated summaries.

• Conduct end-to-end testing for transcription accuracy, summarization quality, and system performance.

• Optimize algorithms based on feedback from test results.

**4. Final Testing, Deployment & Documentation (1.5 weeks)**

**•** Comprehensive system testing to validate functionality, accuracy, and robustness under different scenarios.

• Deployment to a production or demo environment.

• Documentation for users and developers, including a system guide, API references, and troubleshooting steps.

Team Roles & Contributions

This project encompasses product management, cloud architecture, AI development, frontend and backend development, UI/UX design, and quality assurance. Successful implementation relies on a multifunctional team, with each member contributing in their respective roles:

* Ximeng Liu (Team Leader & QA Engineer): Oversees team management, task coordination, quality assurance, and testing, ensuring product stability and feature completeness.
* Rongze Gao (Product Owner): Manages product requirements, defines feature priorities, and collaborates with the team to align the product with market needs.
* Jamie Zhang (Cloud Engineer): Designs and deploys cloud infrastructure, optimizes system performance and scalability, and ensures project stability.
* Tianfa Zhu (UI/UX Designer): Designs the user interface and interaction experience, enhancing the usability and user-friendliness of the meeting assistant.
* Tianxiang Zhang (Software Developer): Handles frontend and backend development, implementing core functionalities and ensuring seamless business logic execution.
* Yuheng Li (AI Engineer): Develops speech-to-text and AI summarization algorithms, optimizing meeting content analysis and improving system intelligence.
* Zining He (Software Architect): Designs the system architecture, establishes technical standards, and ensures software stability and scalability.

The team follows an Agile (Scrum) development process, progressing in iterative phases with regular progress assessments to ensure efficient collaboration and continuous improvement, ultimately delivering a high-quality product.

**Sprint Plan & Progress Tracking**

This project follows an Agile development approach, divided into multiple Sprint iterations, each lasting three weeks. Through short development cycles, we continuously deliver usable features, identify issues early, and adjust plans accordingly. Below is the planned Sprint breakdown and goals for each phase:

**Sprint 1: Planning & Foundation Setup**

**Timeline:** Weeks 3–6  
**Goal:** Finalize requirements, design system architecture, and set up the project framework.

**Tasks:**

* Confirm detailed requirements (e.g., transcription accuracy goals, summary format).
* Select speech recognition service (e.g., Google Speech API, Whisper) and preliminary NLP model.
* Set up the backend framework & database schema, implement basic API health checks.
* Build the frontend structure, preparing the UI for real-time transcription display.

**Acceptance Criteria:**

* A project plan & architecture document is delivered.
* Development environment is ready, with a basic backend-frontend integration demo running.

**Sprint 2: Speech-to-Text Implementation**

**Timeline:** Weeks **7–9**  
**Goal:** Enable real-time transcription and display meeting audio as text on the frontend.

**Tasks:**

* Develop the audio input module, integrating the ASR engine for real-time transcription.
* Implement real-time subtitle display on the frontend, including scrolling and speaker differentiation.
* Implement basic speaker diarization logic (e.g., temporary speaker tagging).
* Write unit tests to simulate audio input and verify transcription accuracy.

**Acceptance Criteria:**

* The system can transcribe live audio and display it on the frontend in real-time, meeting accuracy and latency expectations.

**Sprint 3: Initial Summarization Implementation**

**Timeline:** Weeks **10–12**  
**Goal:** Develop a basic AI-powered meeting summarization module.

**Tasks:**

* Implement an extractive summarization model (e.g., TextRank) to generate concise meeting notes.
* Develop a frontend page for displaying post-meeting summaries.
* Implement individual speaker summaries, extracting key points from each participant’s contributions.
* Conduct small-scale testing to evaluate summary quality and coverage, gathering feedback for improvements.

**Acceptance Criteria:**

* After a meeting, the system generates an automatic summary, including key decisions and participant contributions.

**Sprint 4: Feature Enhancement & Optimization**

**Timeline:** Weeks **1–3** (Second Semester)  
**Goal:** Improve system intelligence, usability, and completeness.

**Tasks:**

* Optimize speech recognition parameters to improve transcription accuracy and reduce filler words.
* Upgrade summarization models (e.g., integrate generative AI) to refine meeting notes, extract key decisions & action items.
* Enhance speaker summarization logic, ensuring accurate representation of individual viewpoints.
* Improve frontend usability, adding features like summary editing, export to Markdown/PDF, and real-time meeting annotations.
* Perform full system testing, including stress testing and edge case handling.

**Acceptance Criteria:**

* Core features function smoothly, transcription accuracy exceeds 90%, and summaries accurately capture meeting content.
* User experience is refined based on iterative improvements.

**Sprint 5: Performance & Security Optimization**

**Timeline:** Weeks **4–6** (Second Semester)  
**Goal:** Enhance system performance, security, and scalability.

**Tasks:**

* Further optimize ASR models to improve accuracy in noisy environments.
* Strengthen data security & encryption, ensuring compliance with privacy regulations.
* Improve system performance, reducing latency and increasing concurrent meeting support.
* Conduct large-scale user testing, gather feedback, and fix critical issues.

**Acceptance Criteria:**

* The system performs well under noisy conditions and supports multiple simultaneous meetings.
* All meeting data is securely encrypted, ensuring compliance with security standards.

**Sprint 6: Final Deployment & Documentation**

**Timeline:** Weeks **7–9** (Second Semester)  
**Goal:** Finalize the project for deployment and release.

**Tasks:**

* Fix remaining bugs and fine-tune system performance.
* Deploy the demo version for product validation, ensuring correct configuration and monitoring.
* Prepare user training materials and technical documentation (e.g., Quick Start Guide, System Architecture Overview).
* Conduct a retrospective to document key lessons learned for future iterations.

**Acceptance Criteria:**

* The project passes final validation, documentation is complete, and the team aligns on next-phase plans (if applicable).

By following this Agile framework, the team ensures transparency, adaptability, and continuous progress, allowing the project to evolve toward its final goal of an AI-powered meeting assistant.