Analysis Report

1. Introduction

This Analysis Report is prepared as part of the TOBB University of Economics and Technology 2025–2026 Spring Term graduation project and focuses on examining the problem domain addressed by the proposed AI-powered mock interview system. The purpose of this report is to analyze the challenges faced by new graduates and early-career candidates during interview preparation, evaluate existing approaches, and justify the design choices of the proposed solution.

Interviews play a critical role in early career development, yet many candidates struggle due to limited access to realistic practice environments and objective feedback. While candidates may possess the necessary technical or domain knowledge, they often lack awareness of how effectively they communicate, manage time, or maintain focus during interviews. Traditional preparation methods, such as self-practice or informal mock interviews, typically fail to provide structured evaluation or measurable insight into these aspects.

This report analyzes interview preparation from both a content and behavioral perspective, emphasizing the importance of structured feedback and guided practice. By examining user needs, limitations of existing solutions, and feasibility considerations, the Analysis Report provides the foundation for the system design and requirements defined in subsequent project documents.

2. Current system (if any)

Currently, interview preparation for new graduates and early-career candidates is mostly limited to self-study, online question lists, and informal mock interviews. While these methods help candidates become familiar with common questions, they do not provide realistic interview interaction or structured, objective feedback. Existing AI-based tools offer limited evaluation and often focus on question delivery rather than communication behavior and improvement-oriented guidance. As a result, candidates lack a consistent and scalable system that supports repeatable practice with meaningful feedback.

3. Proposed system

3.1 Overview

The proposed system is a web-based AI mock interview platform designed to support interview preparation for new graduates and early-career candidates. The system enables users to practice short, structured HR and Technical (non-coding) interviews in a controlled and repeatable environment, with a target session duration of 10–15 minutes.

The platform operates through a voice-based AI interviewer that interacts with the user in real time. Users configure the interview by selecting the interview type, target role, domain interests, difficulty level, and interviewer behavior mode. Two interviewer modes are supported: Supportive and Neutral. While both modes apply the same evaluation criteria, they differ in interaction style. Supportive Mode provides guidance such as gentle off-topic redirection and assistance when uncertainty is expressed, whereas Neutral Mode maintains a standard professional interview tone.

A central component of the system is the feedback mechanism. After each interview session, the system generates a structured feedback report that combines analysis of answer content with communication and behavioral indicators. Content-level evaluation focuses on relevance, clarity, and completeness of responses, while behavioral analysis considers observable signals such as pauses, filler words, speaking rhythm, focus proxy, head movement, and coarse facial expression cues derived from camera input (with explicit user consent).

The system integrates an LLM component primarily for interview dialogue management and content evaluation, while rule-based controls and guardrails manage interview flow, time constraints, difficulty boundaries, and mode-specific behavior. Behavioral signal extraction is treated as a separate analysis pipeline using open-source audio and computer vision techniques. Optional features such as hint cards, progress tracking, or avatar-based interaction are considered outside the core scope and may be included only if project time permits.

3.2 Functional Requirements

As defined in the Project Specifications Report of this project, the functional requirements are as follows:

FR-01: The system shall allow the user to select an interview type (HR or Technical) before starting a session.

FR-02: The system shall allow the user to provide interview context, including target role/position, company or industry context.

FR-03: The system shall allow the user to select a domain/interest area and a difficulty level suitable for junior and intermediate candidates.

FR-04: The system shall provide two interaction modes: Supportive and Neutral.

FR-05: The system shall conduct a voice-based interview session with a target duration of 10–15 minutes.

FR-06: The system shall capture candidate speech via microphone and convert it to text (STT) to support interview flow and analysis.

FR-07: The system shall generate interview questions and follow-up prompts consistent with the selected context and difficulty level.

FR-08: The system shall enforce time boundaries in all modes by prompting the candidate to summarize when answer duration exceeds the allocated window.

FR-09: The system shall, in Supportive Mode, gently steer the candidate back to the question when answers drift off-topic.

FR-10: The system shall, in Supportive Mode, provide supportive guidance when the candidate expresses uncertainty (e.g., "I don't know / I'm not sure") using clarifying sub-questions and small hints.

FR-11: The system shall generate a post-session feedback report including scores and actionable recommendations based on both content quality and communication/behavioral signals.

FR-12: The system shall provide interviewer speech output using text-to-speech (TTS).

FR-13: The system shall use camera input (with explicit user consent) to extract behavioral signals for feedback (e.g., focus/engagement proxy, head movement, and coarse facial expression cues).

3.3 Nonfunctional Requirements

As defined in the Project Specifications Report of this project, the nonfunctional requirements are as follows:

NFR-01 Usability: The system shall provide a simple, guided flow enabling users to start an interview in a small number of steps.

NFR-02 Responsiveness: The system shall provide an interactive experience with conversational responses produced within a reasonable time for live usage.

NFR-03 Reliability: The system shall complete the interview and provide feedback even if some non-critical components degrade (e.g., partial loss of specific behavioral signals), while maintaining the core voice-based flow.

NFR-04 Privacy and security: The system shall require explicit consent for microphone/camera usage and shall not expose API credentials in the client application.

NFR-05 Maintainability: The system shall be modular to allow optional features to be added or removed without impacting the core interview flow.

3.4 Pseudo requirements

As defined in the Project Specifications Report of this project, the pseudol requirements are as follows:

PSR-01 Guided interview flow: The system is expected to maintain a coherent interview progression, ensuring that questions, follow-up prompts, and timing interventions support a realistic interview experience.

PSR-02 Mode-consistent behavior: The interviewer's interaction style is expected to remain consistent with the selected mode (Supportive or Neutral) throughout the session, while applying the same evaluation criteria in both modes.

PSR-03 Balanced guidance: In Supportive Mode, guidance behaviors such as redirection or clarification are expected to assist the user without interrupting the interview flow or revealing direct answers.

PSR-04 Controlled AI behavior: AI-driven components are expected to operate within predefined scope boundaries, including difficulty level, topic relevance, and session timing, through rule-based guardrails.

PSR-05 Transparent feedback presentation: Feedback is expected to be presented in a clear and interpretable manner, allowing users to understand how scores and observations relate to their interview performance.

PSR-06 Non-intrusive behavioral analysis: Behavioral signal extraction is expected to operate unobtrusively during the interview and be used solely for post-session feedback, subject to user consent.

PSR-07 Graceful degradation: If certain analysis components become unavailable or produce incomplete data, the system is expected to continue the interview and provide feedback based on available information.

3.5 System models

This section presents abstract system models used to describe the structure and behavior of the proposed system. The models aim to clarify how users interact with the system, how major components relate to each other, and how the system behaves during an interview session. These models are intended for analysis and understanding rather than detailed implementation.

3.5.1　Scenarios

- **Scenario 1: Standard Interview Session (Neutral Mode)**

A user accesses the platform and selects an interview type (HR or Technical), target role, domain interests, and difficulty level. The user chooses Neutral Mode and starts the interview. The system presents questions via voice output, captures spoken responses, and manages timing. After the session ends, the system generates a feedback report summarizing content quality and communication indicators.

- **Scenario 2: Supportive Interview Session with Guidance**
  A user configures an interview in Supportive Mode. During the interview, the user provides an off-topic response or expresses uncertainty. The system gently redirects the user or offers clarifying prompts without interrupting the interview flow. Time-bound intervention is applied when necessary. A structured feedback report is generated at the end of the session.

- **Scenario 3: Interview Session with Behavioral Analysis**
  A user consents to camera access before starting the interview. During the session, the system analyzes speech and visual signals to extract behavioral indicators. These indicators are included in the post-session feedback alongside content evaluation.

3.5.2 Use case model

This section describes the main use cases of the proposed system by explaining how the user interacts with the system during an interview session. The use case model focuses on user-visible system behavior and defines the boundaries of system functionality without referring to implementation details. Primary Actor is a user who represents a candidate who uses the system to practice HR or Technical interviews and receive feedback.

- **Use Case 1: Configure Interview Session**

  In this use case, the user prepares the interview session by selecting the interview type (HR or Technical), target role or position, relevant domain or interest areas, difficulty level suitable for junior or intermediate candidates, and the interviewer behavior mode (Supportive or Neutral). This use case establishes the configuration parameters that guide the interview flow.

- **Use Case 2: Grant Audio/Video Consent**

  Before the interview begins, the user is prompted to grant explicit consent for microphone access and camera access. Microphone consent is required for the interview to proceed, while camera consent enables behavioral analysis. The user may choose to grant or deny camera access without preventing the interview from taking place.

- **Use Case 3: Participate in Mock Interview**

This is the core use case of the system. The user participates in a voice-based interview session where the system presents questions and the user responds verbally. The system manages question progression, captures and transcribes responses, and monitors session timing.

- **Use Case 4: Receive Supportive Guidance (Extension)**

  This use case extends the "Participate in Mock Interview" use case and is activated only when Supportive Mode is selected. If the user's response drifts off-topic or the user explicitly expresses uncertainty (e.g., "I don't know" or "I'm not sure"), the system provides gentle redirection or clarifying prompts to support the user without interrupting the interview flow.

- **Use Case 5: Time-Bound Prompting**

  This use case is included within the interview flow for both interaction modes. When the user exceeds the allocated response time, the system prompts the user to summarize and return to the main point. This behavior helps keep the interview within the predefined time limits.

- **Use Case 6: View Feedback Report**

  After the interview session is completed, the user views a structured feedback report generated by the system. The report includes content-level evaluation (e.g., relevance, clarity, completeness) and communication and behavioral indicators derived from speech and camera-based analysis, presented in a clear and interpretable format.

3.5.3   Object and class model

The updated object and class diagram illustrates the core conceptual entities of the AI-powered mock interview platform and the high-level relationships between them. It provides a structural view focused on responsibilities and interactions, highlighting where external AI services are integrated through an adapter layer, without introducing implementation-level details.

At the top level, the User class represents the individual interacting with the system. A user may initiate zero or more InterviewSession instances over time. The User's primary actions are to participate in interview sessions and view the generated feedback.

The InterviewSession class remains the central coordinating entity. It encapsulates the configuration of a single interview (e.g., interview type, target role/position, company/industry context, domain interests, difficulty level, and interaction mode) and manages the session lifecycle (start/end). During a session, it coordinates the

interview flow, question/response tracking, behavioral signal collection, and final feedback production.

Each InterviewSession uses exactly one InterviewerAgent, which represents the interviewer's conversational behavior. The InterviewerAgent is responsible for delivering questions and applying mode-related behaviors (Supportive vs. Neutral), such as supportive guidance and time-bound prompting. This separation enables interviewer behavior to evolve independently from session orchestration and data collection.

To manage session flow and enforce system-level guardrails, the InterviewSession is controlled by a BackendOrchestrator. The BackendOrchestrator is responsible for global flow control decisions such as enforcing time boundaries, applying rule-based constraints, and coordinating calls to AI services through a dedicated service layer. The BackendOrchestrator uses an AIServiceGateway, which centralizes high-level AI service operations (e.g., speech transcription, dialogue generation, and voice synthesis). External AI interactions are abstracted behind the OpenAIClient marked as an adapter, which represents the platform's integration point to OpenAI services. This design makes external AI usage explicit while isolating vendor-specific details from core domain entities.

An interview session contains one or more Question objects representing questions or follow-up prompts presented during the interview. For each question, the session collects one or more Response objects. A Response records the candidate's spoken input, the resulting transcript, and response duration. Each Response is associated with the Question it answers, ensuring traceability between prompts and candidate answers.

Upon completion, exactly one FeedbackReport is produced per InterviewSession. The FeedbackReport aggregates evaluation outputs and provides an overall score and structured feedback. To support evidence-based feedback, the report aggregates zero or more BehavioralSignal objects. These signals represent extracted communication and behavioral indicators. In the updated diagram, behavioral signal extraction is handled by a dedicated BehaviorAnalyzer used by the InterviewSession (with user consent), which produces BehavioralSignal instances (e.g., focus/engagement proxy, head movement patterns, coarse facial expression cues, and speech fluency indicators).

Overall, the updated model emphasizes the InterviewSession as the coordinating entity while making the system's architecture-facing responsibilities explicit: interviewer behavior is encapsulated in InterviewerAgent, session control and guardrails are handled by BackendOrchestrator, external AI integration is isolated via AIServiceGateway and the OpenAIClient adapter, and behavioral scoring is encapsulated in the BehaviorAnalyzer. This structure supports extensibility while keeping the core interview flow stable and clearly separated from external service dependencies.
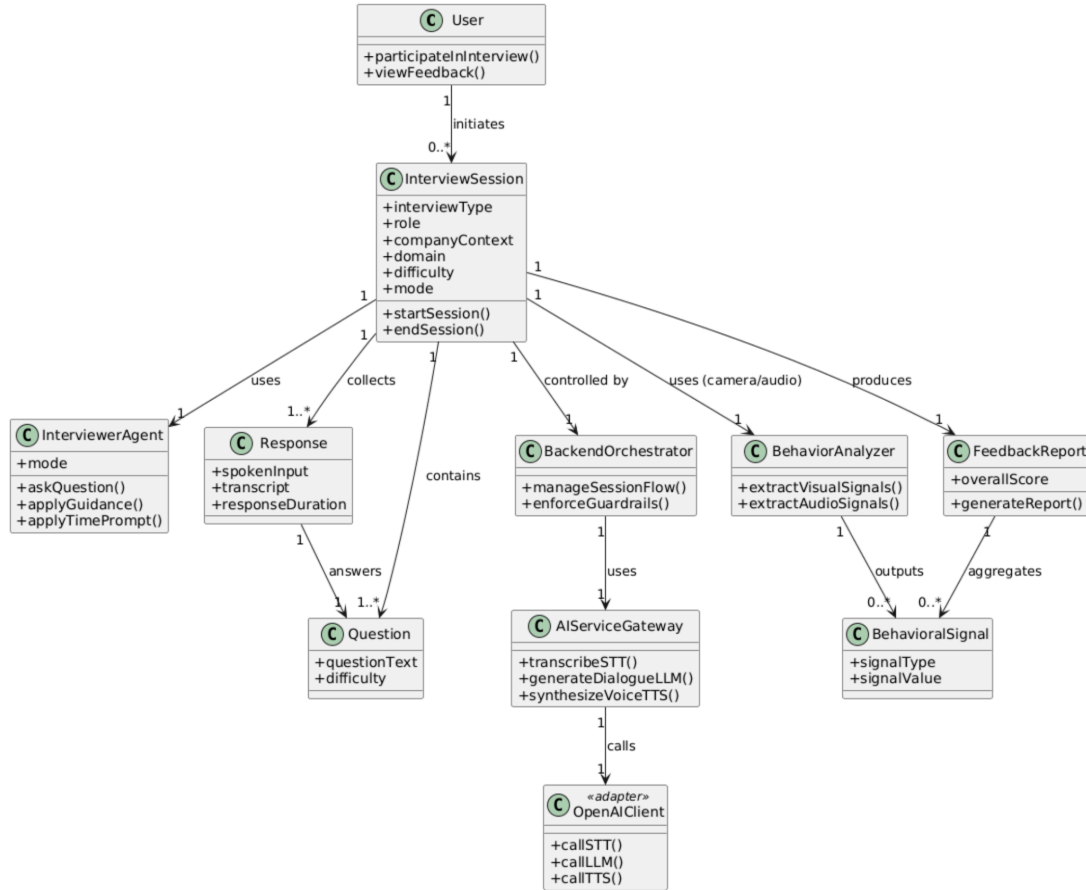
Figure 1. UML Class Diagram

3.5.4 Dynamic models

Dynamic models describe how the system behaves over time and how interactions between system components occur during an interview session. In this project, the dynamic behavior is best represented through a sequence of interactions that take place from the start of an interview until feedback generation.

The dynamic model focuses on the runtime collaboration between the user, the interview session controller, the interviewer agent, and the feedback generation mechanism. It emphasizes the flow of control, data exchange, and mode-dependent behavior without detailing internal algorithms.

Interview Session Interaction Flow

1.   The user starts an interview by initiating an interview session.
2.   The system creates and initializes an interview session based on the selected configuration.
3.   The interviewer agent presents a question to the user.

4.   The user responds verbally to the question.

5.   The system captures the response, transcribes speech, and measures response duration.

6.   During the response:

○   If the response exceeds the allocated time, a time-bound prompt is triggered.

○   If Supportive Mode is active and the response is off-topic or uncertain, supportive guidance is applied.

7.   The process repeats for subsequent questions until the interview session ends.

8.   After completion, the system aggregates responses and behavioral signals to generate a feedback report.

9.   The feedback report is presented to the user.

This dynamic flow can be represented using a sequence diagram to illustrate the temporal order of interactions.

Sequence Diagram

| User | InterviewSession | BackendOrchestrator | AIServiceGateway | OpenAIClient | InterviewerAgent | BehaviorAnalyzer | FeedbackReport |

Session Start

User → InterviewSession: participateInInterview()
InterviewSession: startSession()
InterviewSession → BackendOrchestrator: manageSessionFlow()
BackendOrchestrator: enforceGuardrails()

> Mode selection + sample question preview occur before startSession() in UI.
> Camera stream is captured during session for later analysis (with consent).

Ask First Question

BackendOrchestrator → AIServiceGateway: generateDialogueLLM(context)
AIServiceGateway → OpenAIClient: callLLM()
OpenAIClient → AIServiceGateway: questionText
AIServiceGateway → BackendOrchestrator: questionText
BackendOrchestrator → InterviewerAgent: askQuestion(questionText)
InterviewerAgent → BackendOrchestrator: Question(questionText, difficulty)
BackendOrchestrator → AIServiceGateway: synthesizeVoiceTTS(questionText)
AIServiceGateway → OpenAIClient: callTTS()
OpenAIClient → AIServiceGateway: audio
AIServiceGateway → BackendOrchestrator: audio
BackendOrchestrator → InterviewSession: deliver question audio/text

Q&A Loop (NO behavior extraction here)

loop [Until time/turn limit]

User → InterviewSession: spoken answer (mic) + camera stream (recorded for later)
BackendOrchestrator → AIServiceGateway: transcribeSTT(spokenInput)
AIServiceGateway → OpenAIClient: callSTT()
OpenAIClient → AIServiceGateway: transcript
AIServiceGateway → BackendOrchestrator: transcript
BackendOrchestrator: enforceGuardrails() (time window / silence)

alt [Supportive Mode]

  alt [Answer time exceeded (Supportive)]
  BackendOrchestrator → InterviewerAgent: applyTimePrompt()
  InterviewerAgent → BackendOrchestrator: time prompt delivered

  BackendOrchestrator → InterviewerAgent: applyGuidance() (off-topic or uncertainty)
  InterviewerAgent → BackendOrchestrator: guidance delivered

[Neutral Mode]

  alt [Answer time exceeded (Neutral)]
  BackendOrchestrator → InterviewerAgent: applyTimePrompt()
  InterviewerAgent → BackendOrchestrator: time prompt delivered

> No coaching guidance (same scoring rubric)

BackendOrchestrator → AIServiceGateway: generateDialogueLLM(next question)
AIServiceGateway → OpenAIClient: callLLM()
OpenAIClient → AIServiceGateway: nextQuestionText
AIServiceGateway → BackendOrchestrator: nextQuestionText
BackendOrchestrator → InterviewerAgent: askQuestion(nextQuestionText)
InterviewerAgent → BackendOrchestrator: Question(nextQuestionText, difficulty)

End Session

InterviewSession → BackendOrchestrator: endSession()

Post-Session Behavior Analysis

BackendOrchestrator → BehaviorAnalyzer: extractVisualSignals(recorded camera)
BehaviorAnalyzer → BackendOrchestrator: BehavioralSignal (visual metrics)
BackendOrchestrator → BehaviorAnalyzer: extractAudioSignals(recorded audio)
BehaviorAnalyzer → BackendOrchestrator: BehavioralSignal (audio metrics)

Report Generation

BackendOrchestrator → FeedbackReport: generateReport() (aggregate content + behavior)
FeedbackReport → BackendOrchestrator: overallScore + report
BackendOrchestrator → User: viewFeedback(report)

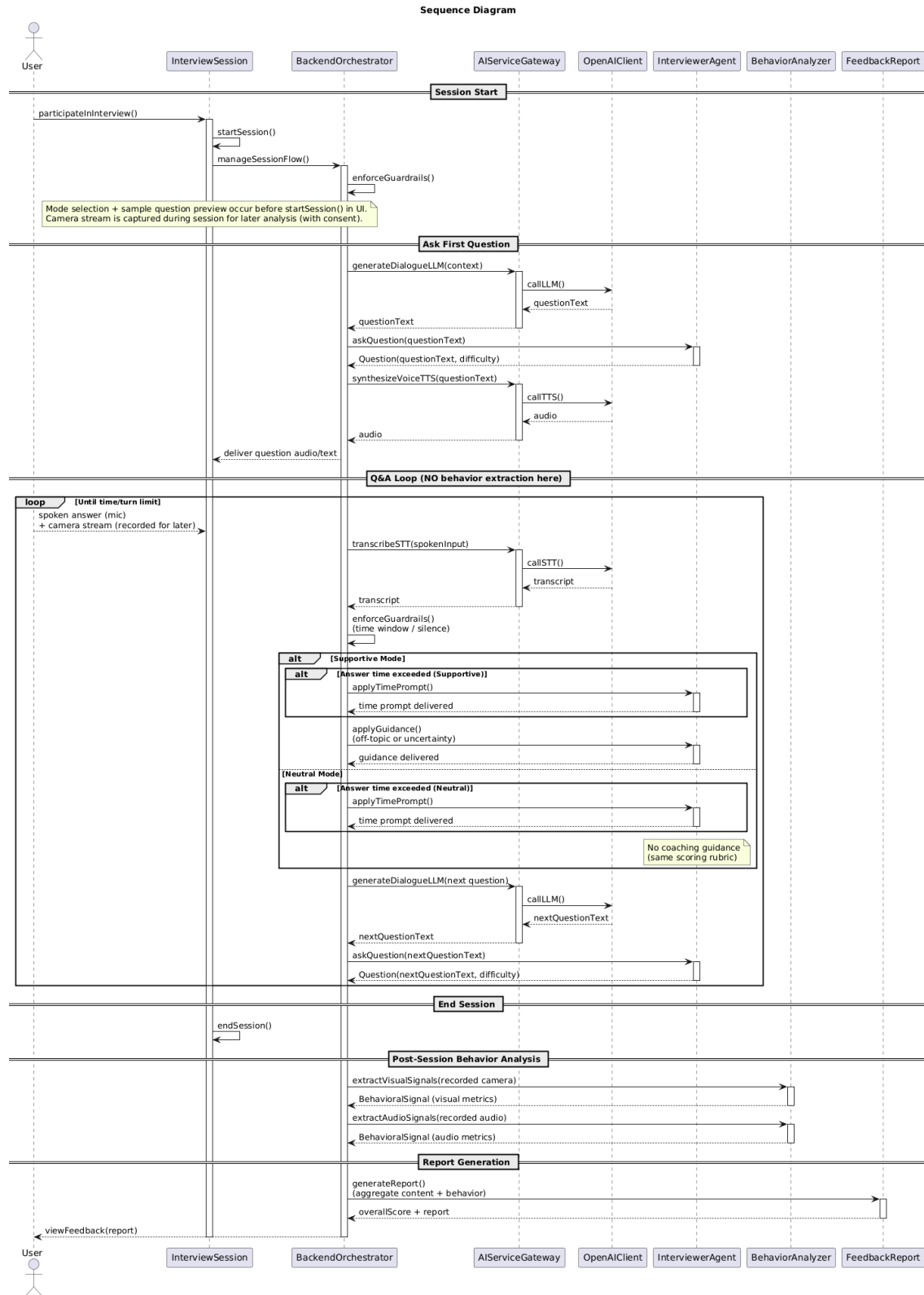| User | InterviewSession | BackendOrchestrator | AIServiceGateway | OpenAIClient | InterviewerAgent | BehaviorAnalyzer | FeedbackReport |

Figure 2. Sequence Diagram of a Single Interview Turn

3.5.5 User interface - navigational paths and screen mock-ups

The primary navigation follows a sequential path consisting of three main stages:

1. Session Setup Screen:
   The user begins by configuring the interview session. This screen allows the user to select the interview type (HR or Technical), target role or position, domain or interest areas, difficulty level, and interaction mode (Supportive or Neutral). Consent options for microphone and camera access are also presented at this stage. Once the configuration is completed, the user proceeds to start the interview.

2. Interview Screen:
   This screen supports the core interaction. The current interview question is displayed in text form while being delivered through voice output. The interface provides visual indicators for session progress and remaining time, but avoids unnecessary distractions. User interaction is primarily voice-based, with minimal manual input required during the interview.

3. Feedback Screen:
   After the interview session ends, the user is redirected to the feedback screen. This screen presents performance scores, summarized observations, and improvement suggestions in a structured and readable layout. Navigation from this screen allows the user to either exit the system or initiate a new interview session.

Screen Mock-ups:

At the analysis stage, screen mock-ups are conceptual rather than detailed. Each screen is defined by its functional purpose rather than specific visual elements or layouts. The mock-ups serve to illustrate the flow of information and user actions across screens, rather than prescribing final user interface designs.

Optional screens such as login, dashboards, or avatar-based interviewer views may be introduced in later development stages if time permits. These optional elements are not considered part of the core user interface flow and do not affect the primary navigation path described above.

4. **Glossary**

- **Behavioral Signals:** Measurable indicators derived from camera and audio processing (e.g., head movement patterns, focus proxy, coarse facial expression cues).
- **Camera-Based Behavioral Analysis:** The extraction of observable communication cues from camera input to support post-session feedback (used only with explicit consent).

- **Content Evaluation:** Assessment of the quality of the candidate's answers based on transcript, such as relevance, clarity, and completeness.
- **Focus / Engagement Proxy:** An approximate indicator of attentiveness inferred from camera-based cues (e.g., consistent face presence and head orientation).
- **Guardrails:** Application-level rules that constrain AI behavior and keep the interview within predefined boundaries (e.g., scope, difficulty, timing, and safety).
- **Interviewer Modes:** Two interaction styles of the interviewer: Supportive (guidance-oriented) and Neutral (standard professional style).
- **Off-Topic Detection:** Identification of responses that deviate from the question's intent, used to trigger supportive redirection when applicable.
- **Time-Bound Prompting:** A system behavior that prompts the candidate to summarize when the response exceeds the allocated time window (applies in all modes).
- **Uncertainty Assistance:** Supportive guidance triggered when the candidate expresses uncertainty (e.g., "I don't know / I'm not sure") using clarifying prompts.
- **Filler Words:** Non-content speech fillers indicating hesitation (e.g., "ııı," "eee," "şey"), used as a fluency-related signal.
- **MVP (Minimum Viable Product) :** The first functional version of the system that includes the core features required for demonstration and evaluation.
- **KVKK:** Turkey's Personal Data Protection Law (Law No. 6698), referenced for consent and privacy-aware handling of user data.

5. References

Project Specifications Report