**Mutah University – Graduation Project Team**

| **Meeting Title** | **Technical Review and Instructor Alignment** |
| --- | --- |
| **Date** | 06/08/2025 |
| **Time** | 09:40PM – 10:30PM |
| **Location** | Our Discord Server |
| **Prepared By** | Faisal Albaba |

**Attendees:**

| **Name** | **Role** |
| --- | --- |
| Faisal Albaba | Team Member |
| Nart Apish | Team Member |
| Qotiph Albayaydah | Team Member |
| Shahem Abo-Alheja | Team Member |
| Bara’a Quneibie | Team Member |

**Agenda:**

1. Discuss Quiz AI Idea Outlines
   * Problem it solves, target users, input types, quiz types.
2. Create Abstract for the Project Documentation
3. Define Technology Stack & Architecture
4. MVP: Upload content → Generate quiz → Solve quiz → Get feedback.

**Discussion Points & Decisions:**

1. **Discuss Quiz AI Idea Outlines**

The Quiz AI system begins with a user login, ensuring secure access to personalized content. The core idea revolves around converting educational materials into interactive quizzes. Supported input formats include MP3 and MP4 for audio/video content, as well as PPTX, DOCX, and PDF for text-based files. The system processes this content to generate various question types, such as multiple choice (MCQ), fill-in-the-blank, mention-based questions, and true/false. The website operates in English and, upon quiz completion, immediately displays the user’s result. Additionally, each user has a private library where all generated quizzes are stored for future review and practice.

1. **Define Technology Stack & Architecture**

The Quiz AI system is designed as a web application using a multi-language architecture. The frontend is built with React.js to provide a dynamic and responsive user interface. The backend is developed using C# (.NET Core Web API), responsible for handling user authentication, file uploads, data management, and communication with the AI module. For AI processing tasks—such as speech-to-text conversion, natural language understanding, and quiz generation—a Python microservice (using FastAPI or Flask) is integrated via REST APIs. Files uploaded by users are stored in a file system or cloud storage, while metadata and quiz data are stored in a relational SQL database. This architecture ensures modularity, scalability, and smooth interaction between system components.

1. **MVP: Upload content → Generate quiz → Solve quiz → Get feedback.**

The Minimum Viable Product of **Quiz AI** focuses on delivering the core functionality needed to validate the project’s main objective. In this version, users can upload educational content in various formats, including audio, video, or text. The AI module then processes the input to automatically generate quizzes based on the material. Users can take the quiz through the web interface, and upon completion, they receive immediate feedback and results. This MVP version also includes user login and the ability to save generated quizzes in a personal library for later review. It provides a complete learning loop from input to evaluation, proving the system’s practical value with minimal complexity.

**Action Items:**

| **Task** | **Responsible** | **Deadline** |
| --- | --- | --- |
| research and prepare security features, including OAuth authentication | Shahm | 10/08/2025 |
| prepare the MoM for the current meeting and presentation for the upcoming meeting | Faisal | 08/08/2025 |

**Next Meeting:**

* **Date & Time**: 08/08/2025
* **Agenda for Next Meeting**:

1. Present and validate the core idea of the Quiz AI project.
2. Confirm supported input types: MP3, MP4, PPTX, DOCX, and PDF.
3. Review the types of questions generated:  
   • Multiple Choice (MCQ)  
   • Fill-in-the-blank  
   • Mention-based questions  
   • True/False
4. Explain the proposed technology stack:  
   • Frontend: React.js  
   • Backend: C# (.NET Core)  
   • AI Microservice: Python
5. Discuss the MVP scope:  
   • Upload content → Generate quiz → Solve quiz → Get feedback
6. Show system features: English language support, result display, and user-specific quiz libraries.