An Intelligent Video-Based Question Answering Framework Using Deep Multimodal Learning

Apeksha H H*, Prajwal R Kolekar *, Vinuth A D*, Ranjan V**
*Students, VII Sem, Dept. of AIML, Jawaharlal Nehru New College of Engineering, Shivamogga
**Assistant Professor, Dept. of AIML, Jawaharlal Nehru New College of Engineering, Shivamogga

***Abstract*—The rapid growth of video-based learning platforms has created a need for automated tools that can transform unstructured multimedia content into assessable educational resources. This paper presents an Artificial Intelligence–driven Video-Based Multiple Choice Question Answering (MCQ-QA) system that automatically generates quizzes from educational videos. The proposed framework accepts a YouTube video link as input and extracts the corresponding audio stream, which is transcribed into textual content using the Whisper speech-to-text model. The generated transcript is then processed by a generative AI model to create contextually relevant multiple-choice questions, along with answer options, correct responses, and explanatory feedback.**

The complete workflow is implemented within a web-based application developed using Streamlit, offering an interactive interface for quiz participation and real-time feedback. Visualization of learner performance is enabled through graphical analysis, while automated report generation produces downloadable PDF summaries containing questions, answers, and assessment results. The proposed system significantly reduces the manual effort required for question paper preparation, enhances scalability of learning assessments, and supports personalized self-evaluation. Experimental validation demonstrates the system's effectiveness in converting video lectures into structured, high-quality assessment material. This work highlights the potential of AI-driven automation in enriching video-based education and intelligent learning analytics.

***Keywords*— Video-Based Question Answering; Automated Quiz Generation; Speech-to-Text; Generative Artificial Intelligence; Educational Technology; Learning Assessment; Natural Language Processing; AI in Education.**

I. INTRODUCTION

The Video Based Multiple Choice Question Answer System using AI is designed to convert educational video content into automatically generated assessments. The system begins by accepting a YouTube video link, extracting its audio using yt-dlp, and converting the speech into text through the Whisper speech recognition model. This ensures accurate transcription of spoken content, which forms the foundation for quiz generation.

The transcribed text is processed using a generative AI language model to create multiple choice questions that include four options, correct answers, and short explanations. A user friendly interface built with Streamlit allows learners or instructors to view generated questions, attempt the quiz, and evaluate their responses. The system displays feedback instantly and visualizes performance using charts to make evaluation simple and clear.

To support reporting and documentation, the project integrates PDF generation through ReportLab and creates downloadable text files. Users can download both the generated questions and their assessment results for reference or record keeping. Overall, this system reduces manual workload in question creation, enhances digital learning evaluation, and transforms video-based content into structured, measurable learning outcomes.

With the rapid growth of online educational platforms, video-based learning has become a primary mode of knowledge delivery. However, assessing learners' understanding of video content remains a largely manual, time-consuming, and instructor-dependent process. Educators must repeatedly watch videos, identify key concepts, and manually design quizzes or assessments, which limits scalability and timely feedback. Existing automated assessment systems often fail to directly process video content or lack the ability to generate meaningful, context-aware multiple-choice questions with explanations. Hence, there is a critical need for an intelligent system that can automatically extract knowledge from educational videos and convert it into structured assessments. This project addresses

this gap by proposing an AI-driven solution that transforms video-based learning material into automatically generated multiple-choice questions, enabling efficient, scalable, and objective evaluation of learner understanding while reducing instructor workload and enhancing the overall learning experience.

II. RELATED WORK

Recent advances in video-based learning have increased the demand for tools that can convert lecture videos into structured assessments automatically. Early research on automatic question generation (QG) from video typically followed a pipeline approach: extract audio, transcribe to text, apply linguistic processing (tokenization, POS tagging, NER, dependency parsing), and generate questions using predefined templates such as *Who/What/Where/When/Why/How*. Such systems are simple and interpretable, and they work reasonably well for factual questions derived from clean transcripts. However, literature consistently reports that audio quality and ASR errors propagate downstream, causing entity extraction failures and poor question formulation, while template-based generation limits diversity and higher-order reasoning questions.

To overcome the limitations of rule-driven QG, modern work has adopted transformer-based models for summarization and question generation. Frameworks like VideoDL emphasize end-to-end automation with a teacher-in-the-loop design: video transcription is followed by segmentation and summarization (e.g., using T5), keyword extraction, and multi-format question generation (SAQ, BLQ, GFQ, MCQ). This line of research shows that combining powerful text generation with educator review improves trust, pedagogical alignment, and practical usability. Yet, even in these systems, key challenges remain: keyword selection strongly impacts quality; transcription noise reduces coherence; and automatically produced answers—especially for logical or short-answer formats—may be incorrect in a notable fraction of cases, requiring human validation.

Parallel to QG systems, research on Video Question Answering (VideoQA) provides broader foundations for understanding video and language jointly. Survey literature categorizes VideoQA by question type (factoid vs. inference) and modality (standard, multimodal, knowledge-based), and highlights a common architecture: video encoding, question encoding, cross-modal interaction, and answer decoding. While transformer and graph-based VideoQA models perform strongly on benchmark datasets, surveys note a gap between curated datasets and real-world videos where language bias, scene complexity, and temporal reasoning demands reduce generalization. For education-focused systems that rely mostly on transcript text, these studies reinforce that temporal context, segmentation, and context selection are crucial—because a quiz question is only meaningful if it is anchored to the correct portion of the lecture.

A complementary stream of research investigates the pedagogical impact of quizzes embedded in videos. Empirical studies comparing "no quiz," "quiz at end," and "quiz embedded throughout" formats generally conclude that embedded quizzes improve short-term retention, engagement, and learner feedback—supporting the motivation for automated video-to-quiz systems. However, these studies also caution that results may be influenced by confounds (e.g., quiz similarity to tests, unequal time gaps before assessment) and often lack longitudinal tracking, indicating that automated quiz generation should ideally align with sound instructional design rather than only extracting surface facts.

With the emergence of Large Language Models (LLMs), recent work specifically explores MCQ generation from transcripts and systematic quality evaluation through item-writing flaws (IWFs) such as weak distractors, missing context, ambiguity, and cueing. Findings suggest LLMs can rapidly scale MCQ creation, but quality varies widely: human experts often rate a larger portion as acceptable than automated evaluators, showing inconsistency in evaluation standards. This motivates hybrid workflows: strong prompting + constraint checks + human review, particularly for high-stakes assessment. For practical deployments, studies also emphasize the need to handle transcript imperfections, domain differences, and the mismatch between "video meaning" (visual + audio) and transcript-only meaning.

Overall, the literature indicates that a robust video-based assessment system should: (i) prioritize reliable transcription and segmentation, (ii) use

transformer/LLM generation for richer question diversity, (iii) include explanation generation and answer validation, (iv) support interactive learner feedback and analytics, and (v) provide exportable reports for educators.

III. SYSTEM ARCHITECTURE

The proposed Video-Based Multiple Choice Question Answer System is organized as a modular, sequential data processing pipeline consisting of five major components. Each component performs a well-defined function and passes its output to the subsequent stage, ensuring efficient and scalable system operation.

1. Presentation and Container Layer (Streamlit)
The Presentation Layer serves as both the frontend interface and the central orchestration layer of the system. Implemented using Streamlit, this layer manages user interaction, application state, and the overall execution flow. It accepts user inputs such as the YouTube video URL and the desired number of MCQs, initiates backend processing steps, and displays progress indicators to keep users informed. The Streamlit framework also handles session management using st.session_state, ensuring smooth transitions between transcription, question generation, and assessment stages. Final outputs—including generated quizzes, performance visualizations, and downloadable reports—are rendered dynamically within this interface.

2. Data Acquisition Layer (yt-dlp)
The Data Acquisition Layer is responsible for extracting the audio content from the user-provided YouTube video. This component uses the yt-dlp library, a robust and widely adopted media extraction tool. Upon receiving the video URL from the presentation layer, it downloads and converts the audio stream into a local MP3 file. To ensure reliability, this layer incorporates retry mechanisms and timeout handling to manage potential network failures or video accessibility issues. The extracted audio file forms the primary input for the transcription stage.

3. Transcription and Feature Extraction Layer (Whisper)
The Transcription Layer converts the extracted audio into textual form using the Whisper speech-to-text model, specifically the small.en variant. This model balances transcription accuracy with computational efficiency, making it suitable for deployment on standard hardware while still delivering reliable results. The downloaded audio file is processed locally or within a containerized environment, and the resulting transcription is stored in memory and optionally saved as a text file. This stage is computationally intensive and benefits from GPU acceleration when available, as it directly influences the quality of downstream question generation.

4. Quiz Generation Layer (Gemini LLM)
The Quiz Generation Layer transforms the transcribed text into structured multiple-choice questions using a Large Language Model (LLM). This system integrates the Google Gemini API (gemini-2.5-flash) to generate context-aware questions, answer options, correct answers, and explanations. To address LLM context window limitations, the transcription is divided into manageable text chunks before being sent to the model. A carefully designed JSON-based prompt ensures that the output follows a strict, machine-readable structure. The generated content is parsed and stored as structured data objects, enabling seamless rendering and evaluation within the application.

5. Output and Persistence Layer (ReportLab and Local Storage)
The Output Layer provides formatted and persistent results to the user. Using the ReportLab library, the system generates two types of downloadable PDF documents: a Quiz PDF, containing only the questions and answer options, and an Assessment PDF, which includes user responses, scores, correct answers, and detailed explanations for attempted questions. Additionally, intermediate artifacts such as transcriptions and MCQ data are temporarily stored as local files to facilitate immediate download. This layer ensures that users can retain, reuse, or distribute the generated assessment materials beyond the live application session.

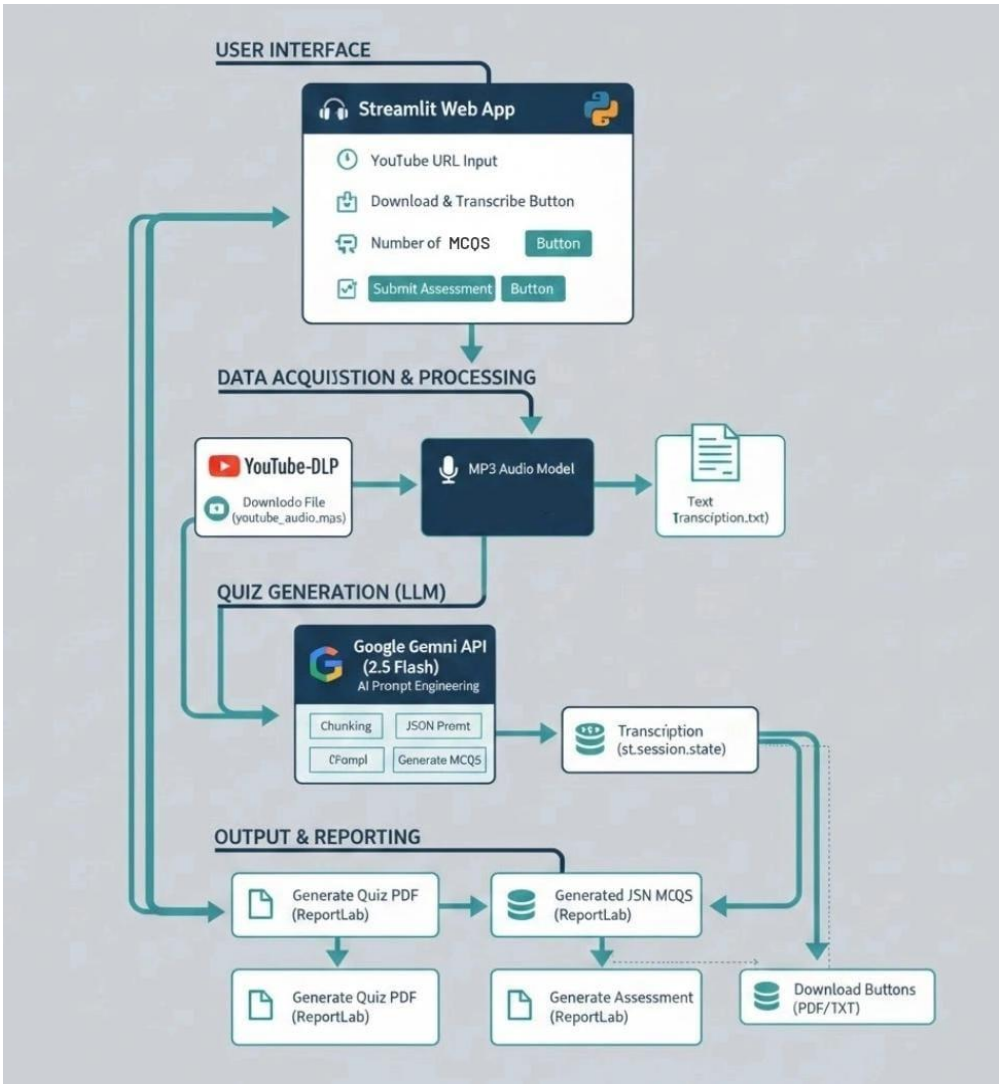

Figure 1: Proposed system

IV. IMPLEMENTATION

Algorithm: AI-Based Video-to-MCQ Generation System
Input:

- YouTube Video URL
- Number of MCQs required

Output:

- Generated MCQs
- Quiz attempt results
- Downloadable PDF report

BEGIN

1. INITIALIZE Streamlit Application
2. DISPLAY input fields for:
   a. YouTube Video URL
   b. Number of MCQs
3. WAIT for user to click "Generate Quiz"

4. IF input URL is invalid THEN
   DISPLAY error message
   TERMINATE process
   END IF

5. DISPLAY "Processing..." spinner

// ---------- AUDIO EXTRACTION ----------
6. CALL extract_audio(video_url)
   a. Use yt-dlp to download audio
   b. Save audio as youtube_audio.mp3
7. IF audio extraction fails THEN
   DISPLAY error message
   TERMINATE process
   END IF

// ---------- SPEECH TO TEXT ----------

```
8. CALL transcribe_audio("youtube_audio.mp3")
    a. Load Whisper model (small.en)
    b. Convert audio to text
    c. Store transcription in session state
    d. Save transcription to file
9. IF transcription fails THEN
    DISPLAY error message
    TERMINATE process
  END IF

// ------------------ TEXT PREPROCESSING ----------
--------
10. CALL preprocess_text(transcription)
    a. Remove unwanted symbols
    b. Normalize spacing
    c. Split text into chunks

// ------------------ MCQ GENERATION ----------------
--
11. INITIALIZE empty MCQ_List

12. FOR each text_chunk in preprocessed_text DO
    a. CALL generate_mcqs(text_chunk)
        i. Send chunk to Gemini LLM
        ii. Request JSON structured MCQs
    b. PARSE JSON output
    c. APPEND MCQs to MCQ_List
    d. IF MCQ_List size >= required_MCQs THEN
        BREAK loop
      END IF
  END FOR

13. STORE MCQ_List in session state

// ------------------ QUIZ DISPLAY ------------------
14. DISPLAY MCQs one by one
15. FOR each MCQ DO
    a. DISPLAY question and options
    b. ACCEPT user answer
    c. STORE user response
  END FOR

// ------------------ EVALUATION ------------------
16. INITIALIZE score = 0
17. FOR each answered MCQ DO
    a. IF user_answer == correct_answer THEN
        INCREMENT score
      END IF
  END FOR

18. CALCULATE percentage_score

// ------------------ VISUALIZATION ------------------
19. CALL plot_results(score, total_questions)
    a. Generate pie chart using Matplotlib
20. DISPLAY chart

// ------------------ REPORT GENERATION ------------
------
21. CALL generate_quiz_pdf(MCQ_List)
22. CALL generate_assessment_pdf(
      MCQ_List,
      user_answers,
      score,
      explanations
    )

// ------------------ OUTPUT ------------------
23. DISPLAY final score
24. PROVIDE download links for:
    a. Quiz PDF
    b. Assessment PDF

25. DISPLAY "Process Completed Successfully"

END
```

V. RESULTS AND DISCUSSIONS

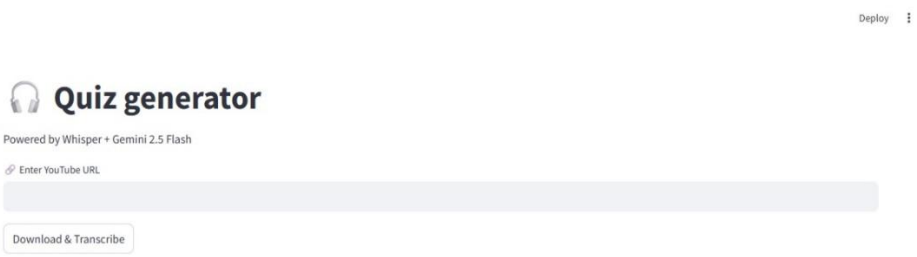

Figure 2: Home Screen

The home screen of the project provides a simple interface where users can start generating quizzes from YouTube videos. It includes a title, a description of the AI technologies used, and a text box to enter the video URL. Once the user enters a link, the Download & Transcribe button initiates automatic audio extraction and transcription. This clean layout makes the process easy and accessible for all users.

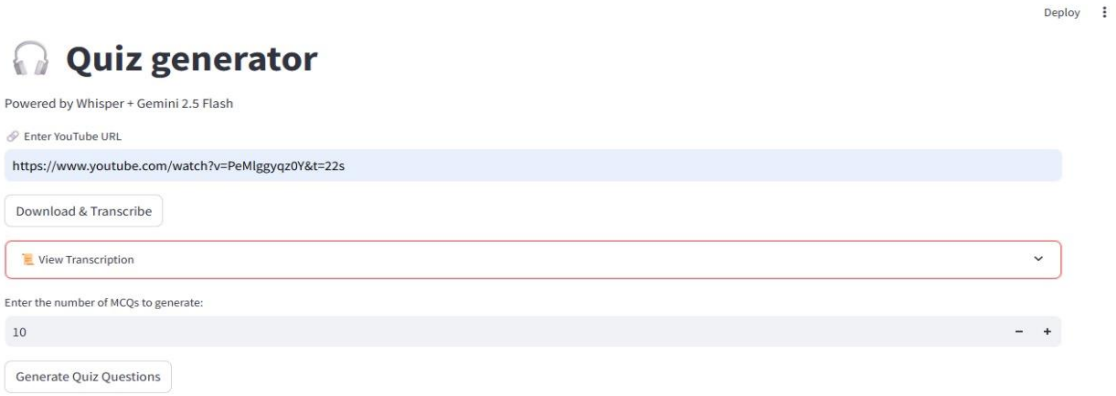

Figure 3: Upload Page

After the YouTube link is entered and transcribed, the interface displays a *View Transcription* option that allows users to check the extracted text. The user can then choose how many MCQs they want the system to generate. This screen provides full control over the quiz creation process, letting users verify the transcription and set the desired number of questions before generating the quiz.

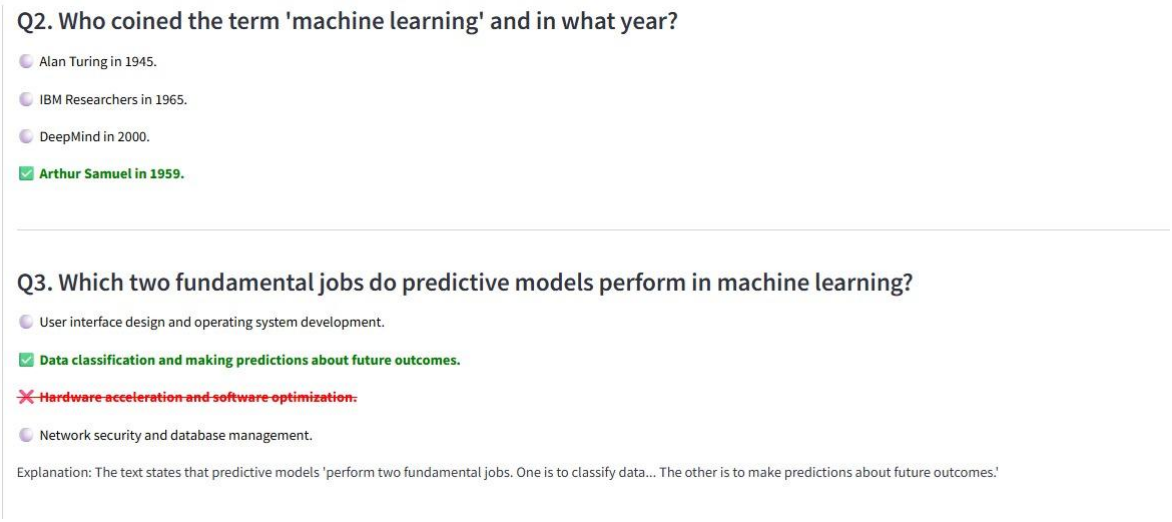

Figure 4: MCQ Result Page

This screen displays the quiz assessment results, showing each question along with the user's selected answer and the correct option. Correct answers are highlighted in green, while incorrect choices are marked in red with a strike-through for clarity. Each question also includes an explanation to help users understand the concept better. This layout makes reviewing performance easy and supports effective learning.

This pie chart visually represents the user's quiz performance based only on the questions they answered. The green portion indicates the percentage of correct answers, while the red portion represents incorrect responses. This clear comparison helps users quickly understand their accuracy and overall performance. The simple visual format makes evaluation easy and intuitive.

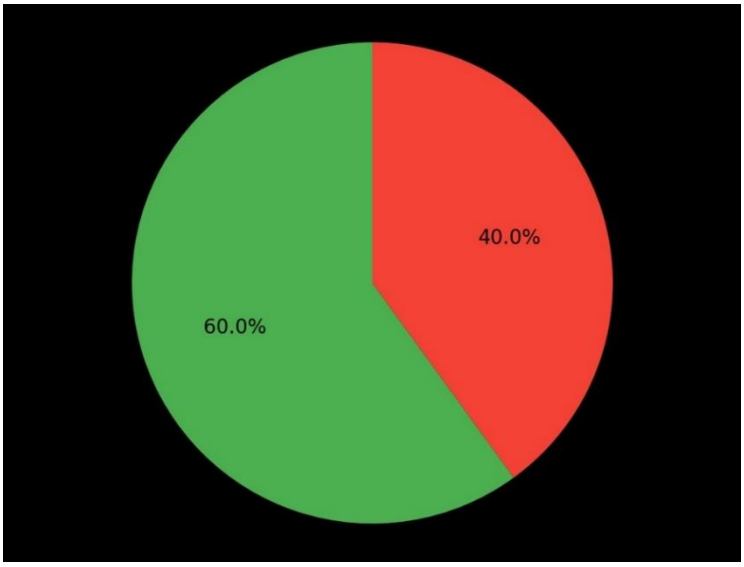

Figure 5: visualization of the result

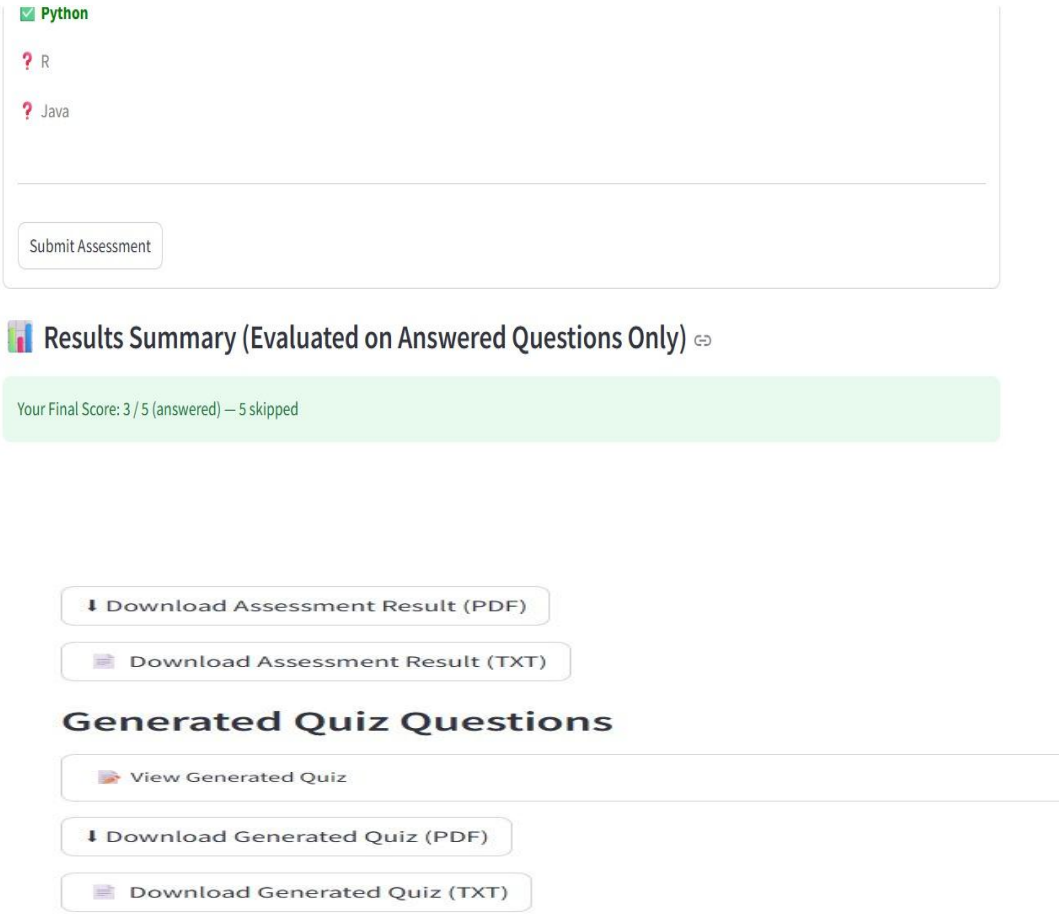

Figure 6: Final Quiz Results & Downloads

This section shows the user's final quiz score, calculated only from the questions they answered, along with the number they skipped. Below the summary, the system provides options to download the assessment results in both PDF and text formats. It also displays the generated quiz questions with buttons to view or download them. This makes it easy for users to save, review, or share their quiz reports and questions.

VI. CONCLUSION AND FUTURE SCOPE

The proposed AI-based Video Quiz Generator system successfully demonstrates a complete end-to-end automated system that transforms YouTube video content into meaningful, interactive assessments. Using yt-dlp for audio extraction, Whisper for accurate speech-to-text conversion, and Gemini 2.5 Flash for generating high-quality MCQs, the system ensures a smooth and intelligent workflow from video input to quiz output.

This work also integrates robust backend logic—such as regex-based JSON extraction, error-handled downloads, dynamic MCQ generation, and only-answered-questions evaluation—to maintain reliability and correctness. The use of Streamlit enables a clean, user-friendly interface where users can watch the system automatically download, transcribe, generate questions, take assessments, and view graphical performance analytics.

Advanced features like PDF generation, shuffled options, partial answer evaluation, pie-chart visualization, and downloadable reports make the application not just functional but practical for real learning environments. Overall, this system has the potential of combining modern AI models with an intuitive UI to automate educational content creation. It is efficient, scalable, and highly useful for teachers, students, and e-learning platforms.

In the future, the system can be expanded to support multilingual transcription and quiz generation, making it useful for diverse educational environments. Advanced AI features such as adaptive difficulty levels, topic-wise video segmentation, and improved distractor generation can further refine the quality of MCQs. Integrating the platform with LMS systems, adding detailed analytics dashboards, and generating automatic summaries or notes from the video can greatly improve the learning experience. Additionally, deploying the application on the cloud, introducing mobile app support, enabling voice-based interaction, and creating a reusable question bank can make the system more scalable and user-friendly. With these advancements, this project can evolve into a powerful, intelligent, and comprehensive e-learning tool.

BIBLIOGRAPHY

[1] B. K. Ajin, R. Riju, E. Sam Edwin, J. Jerome Jebadurai, and A. Shakeela Joy, "Automatic question generation from video using natural language processing," *International Journal on Engineering Technology and Sciences*, vol. 11, no. 3, pp. 28–31, 2024.

[2] P. Rice, P. Beeson, and J. Blackmore-Wright, "Evaluating the impact of a quiz question within an educational video," *TechTrends*, vol. 63, no. 6, pp. 739–748, 2019.

[3] Y. B. Kang, A. R. M. Forkan, P. P. Jayaraman, N. Wieland, E. Kollias, H. Du, S. Thomson, and Y. F. Li, "An AI-based solution for enhancing delivery of digital learning for future teachers," *arXiv preprint arXiv:2112.01229*, 2021.

[4] A. R. M. Forkan, Y. B. Kang, P. P. Jayaraman, H. Du, S. Thomson, E. Kollias, and N. Wieland, "VideoDL: Video-based digital learning framework using AI question generation and answer assessment," *International Journal of Advanced Corporate Learning*, vol. 16, no. 1, pp. 19–27, 2023.

[5] Y. Zhong, J. Xiao, W. Ji, Y. Deng, and T. S. Chua, "Video question answering: Datasets, algorithms and challenges," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022, pp. 6439–6455.

[6] T. Arif, S. Asthana, and K. Collins-Thompson, "Generation and assessment of multiple-choice questions from video transcripts using large language models," in *Proceedings of the 11th ACM Conference on Learning @ Scale*, 2024, pp. 1–10.

[7] M. Merkt, S. Weigand, A. Heier, and S. Schwan, "Learning with videos vs. learning with print: The role of interactive features," *Learning and Instruction*, vol. 21, no. 6, pp. 687–704, 2011.

[8] K. K. Szpunar, N. Y. Khan, and D. L. Schacter, "Interpolated memory tests reduce mind wandering and improve learning of online lectures," *Proceedings of the National Academy of Sciences*, vol. 111, no. 16, pp. 6313–6317, 2014.

[9] R. E. Mayer, *Multimedia Learning*. Cambridge, UK: Cambridge University Press, 2009.

[10] X. Wang et al., "A deep learning model for the prediction of skin cancer," *Artificial Intelligence in Medicine*, vol. 107, pp. 101880, 2020.