SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

Project Report on

"EDU-VIDEO INSIGHT SEARCH, TRANSLATION AND ABSTRACTION"

submitted in partial fulfillment of the requirement for the completion of V semester of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE & ENGINEERING

Submitted by

Khachith Sri Rangu V H (1SI23CS086)

Manoj P J (1SI23CS107)

Naveen Kumar R (1SI23CS120)

Pavan Kumar K J (1SI23CS128)

under the guidance of

Mrs.Tejashwini D A

Assistant Professor

Department of CSE

SIT, Tumakuru-03

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2025-26

SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

Certified that the mini project work entitled as "EDU-VIDEO INSIGHT SEARCH, TRANSLATION AND ABSTRACTION " is a bonafide work carried out by Khachith Sri Rangu V H (1SI23CS086), Manoj P J (1SI23CS107), Naveen Kumar R (1SI23CS120), and Pavan Kumar K J (1SI23CS128) in partial fulfillment for the completion of V Semester of Bachelor of Engineering in Computer Science and Engineering from Siddaganga Institute of Technology, an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year 2025-26. It is certified that all corrections or suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The Mini project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering degree.

Mrs. Tejashwini D A
Assistant Professor
Dept. of CS&E
SIT, Tumakuru–03

Dr. N R Sunitha
Head of the Department
Dept. of CS&E
SIT, Tumakuru–03

External viva:

Names of the Examiners **Signature with date**

1.

2.

ACKNOWLEDGEMENT

We offer our humble pranams at the lotus feet of **His** Holiness, **Dr. Sree Sree Sivakumara Swamigalu**, Founder President and **His** Holiness, **Sree Sree Siddalinga Swamigalu**, President, Sree Siddaganga Education Society, Sree Siddaganga Math for bestowing upon their blessings.

We deem it as a privilege to thank **Dr. Shivakumaraiah**, CEO, SIT, Tumakuru, and **Dr. S V Dinesh**, Principal, SIT, Tumakuru for fostering an excellent academic environment in this institution, which made this endeavor fruitful.

We would like to express our sincere gratitude to **Dr. N R Sunitha**, Professor and Head, Department of CS&E, SIT, Tumakuru for his encouragement and valuable suggestions.

We thank our guide **Tejashwini D A**, Assistant Professor, Department of Computer Science & Engineering, SIT, Tumakuru for the valuable guidance, advice and encouragement.

Khachith Sri Rangu V H (1SI23CS086)
Manoj P J (1SI23CS107)
Naveen Kumar R (1SI23CS120)
Pavan Kumar K J (1SI23CS128)

Course Outcomes

CO1: To identify a problem through literature survey and knowledge of contemporary engineering technology.

CO2: To consolidate the literature search to identify issues/gaps and formulate the engineering problem

CO3: To prepare project schedule for the identified design methodology and engage in budget analysis, and share responsibility for every member in the team

CO4: To provide sustainable engineering solution considering health, safety, legal, cultural issues and also demonstrate concern for environment

CO5: To identify and apply the mathematical concepts, science concepts, engineering and management concepts necessary to implement the identified engineering problem

CO6: To select the engineering tools/components required to implement the proposed solution for the identified engineering problem

CO7: To analyze, design, and implement optimal design solution, interpret results of experiments and draw valid conclusion

CO8: To demonstrate effective written communication through the project report, the one-page poster presentation, and preparation of the video about the project and the four page IEEE/Springer/ paper format of the work

CO9: To engage in effective oral communication through power point presentation and demonstration of the project work

CO10: To demonstrate compliance to the prescribed standards/ safety norms and abide by the norms of professional ethics

CO11: To perform in the team, contribute to the team and mentor/lead the team

Table 0.1: **CO–PO Mapping**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO-1								3		3			3	
CO-2	3		3									3		
CO-3								3		3			3	
CO-4					3	3							3	
CO-5	3	3											3	
CO-6					3								3	
CO-7	3	3	3										3	
CO-8								3					3	
CO-9									3				3	
CO-10					3					3				
CO-11						3							3	

Attainment level: - 1: Slight (low) 2: Moderate (medium) 3: Substantial (high)

POs:

PO1: Engineering Knowledge

PO2: Problem analysis

PO3: Design/Development of solutions

PO4: Conduct investigations of complex problems

PO5: Engineering tool usage

PO6: Engineer and the world, PO7: Ethics

PO8: Individual and collaborative team work

PO9: Communication

PO10: Project management and finance

PO11: Lifelong learning

PSOs:

PSO1: Computer based systems development

PSO2: Software development

PSO3: Computer Communications and Internet applications

ABSTRACT

In the modern digital age, the extensive use of short-form video platforms such as Instagram Reels, YouTube Shorts, and TikTok has had a noticeable impact on students' learning behaviors and attention capacity. Regular exposure to rapid, short-duration content has led to shorter concentration spans and a growing intolerance toward long academic lectures. Research indicates that the average human attention span has declined from approximately 12 seconds in 2000 to nearly 8 seconds in recent years, making it increasingly difficult for learners to maintain focus during online education. Consequently, many students find it challenging to stay attentive throughout full-length instructional videos and to actively engage with the learning content.

Additionally, manual note-taking during lectures interrupts the learning flow, often resulting in missed concepts and requiring students to repeatedly replay video sections. This repetition increases mental workload and stress while negatively affecting overall learning efficiency. These issues emphasize the growing demand for intelligent educational solutions that promote focused and time-efficient learning.

To overcome these challenges, this project introduces an AI-based system designed to automatically transcribe lecture videos, produce clear and concise summaries, and identify essential learning points. By minimizing the need for extensive manual note-taking, the system allows students to focus more effectively on comprehension and critical thinking. The generated content can support rapid revision, customized study resources, and better retention of knowledge. Ultimately, this solution seeks to improve student engagement, reduce cognitive fatigue, and deliver a more effective and accessible digital learning experience.

Contents

Contents i

List of Figures iii

List of Tables iv

Abstract v

1 Introduction 1

1.1 Motivation 1

1.2 Objective of the project 1

1.3 Organisation of the report 2

2 Literature Survey 3

3 System Design & Methodology 7

3.1 Functional & Non-Functional Requirements 7

3.2 List of Hardware & Software Requirements 7

3.3 System architecture 8

3.4 Data Flow diagrams 11

3.5 Algorithms 14

4 Implementation Details 16

5 Results 18

5.1 Screenshots 18

6 Conclusion & Future Enhancement 21

Bibliography 22

Appendices 24

A Self-Assesment of the Project 25

B Sustainable Development Goals addressed 27

C Detailed Test Cases and Results 28

D Implementation Environment and Configuration Details 30

List of Figures

3.1 Layered architecture of the edu-video insight search, translation and abstraction . 11

3.2 workflow model of edu-video insight search, translation and abstraction 12

5.1 E-VISTA Login Page– User authentication interface to securely access the platform 20

5.2 Dashboard Page– Central control panel displaying navigation cards for search, summaries, quizzes, and flashcards 21

5.3 Intelligent Video Search– Search results generated based on user input keywords and queries 21

5.4 Transcript Progress View– Real-time transcription and highlighting of video speech for better comprehension 22

5.5 Structured Summary Generation– Condensed text summaries automatically produced from the video content 22

5.6 Quiz Generation– Auto-created quiz questions to test understanding of summarized concepts 23

5.7 Quiz Results Correct Answers– Displays user responses with correct answer validation and feedback 23

5.8 Flashcards Interface– Key learning points converted into flashcards for quick review and revision 23

List of Tables

0.1 **CO–PO Mapping** ii

5.1 Detailed Functional Test Cases 21

Chapter 1

Introduction

As digital technologies continue to expand across all industries, the importance of cybersecurity has grown significantly in protecting sensitive systems and critical data. Cyber attackers continuously modify their tactics, applying advanced techniques to exploit system vulnerabilities and disrupt essential operations. Conventional firewall systems, which primarily rely on static rule sets and stored attack signatures, are ineffective in identifying newly emerging or highly sophisticated threats that differ from previously observed behavior.

Modern network infrastructures consist of cloud-based services, remotely connected systems, Internet of Things (IoT) devices, mobile platforms, and virtualized environments. This widely distributed architecture increases the number of potential access points available to attackers. Security threats may originate from both internal and external sources, while advanced attack methods such as encrypted malware, zero-day exploits, and automated intrusion tools frequently bypass traditional defense mechanisms. Although Next-Generation Firewalls (NGFWs) provide deeper traffic inspection, they continue to face challenges in detecting unpredictable or unknown attacks in real time.

To overcome these challenges, this project proposes a Machine Learning–powered Firewall Bot (ML-Bot) integrated within an NGFW framework. By utilizing machine learning techniques, the system can examine behavioral patterns, identify anomalies, and automatically respond to malicious activities. When combined with the Zero Trust security model, which requires continuous verification of every access request, this approach enables a more intelligent, proactive, and highly adaptive defense mechanism for securing modern network environments.

1.1 Motivation

Personal Motivation :

Students increasingly depend on extended online video lectures for learning, examination preparation, and skill enhancement. Repeated interruptions for manual note-taking break concentration and result in inefficient use of time. In addition, a large portion of high-quality educational content is available exclusively in English, which creates accessibility challenges for learners who prefer regional languages.

Academic Motivation :

In India, more than 70 million individuals with hearing impairments rely on written transcripts to access educational content. Students, educators, and researchers invest significant effort in manually transcribing lectures, seminars, and interviews. With the rapid expansion of digital education, the demand for automated transcription, summarization, and academic content generation has grown substantially.

Technological Motivation :

Recent progress in artificial intelligence, including models such as OpenAI Whisper for speech recognition and Google Gemini for natural language processing, has made high-accuracy transcription and summarization feasible. This project seeks to combine these advanced technologies into a unified, free, open-source, scalable, and privacy-conscious educational platform.

1.2 Objectives of the E-VISTA

The primary objectives of the E-VISTA system are outlined below:

1. To develop an intelligent search mechanism for educational videos across multiple online platforms.
2. To implement a local AI-driven transcription workflow using Whisper or Faster-Whisper.
3. To produce structured multilingual summaries with the help of Gemini.
4. To automatically generate quizzes and flashcards that improve learner engagement, along with providing multilingual text-to-speech support to enhance accessibility.

1.3 Organisation of the report

This report is organised into multiple chapters, each focusing on a specific aspect of the proposed E-VISTA: AI-Powered Educational Video Analysis Platform, in order to provide a clear and systematic understanding of the project.

Chapter 1 — Introduction: This chapter provides an overview of the project, explaining the background and importance of educational video analysis. It includes the motivation behind the project, objectives, and the scope, highlighting the need for an intelligent system to convert video content into structured learning materials.

Chapter 2 — Literature Survey: This chapter reviews existing research and related work in the fields of educational technology, video summarization, speech-to-text systems, and AI-based learning platforms. It compares previous approaches with the proposed system and identifies the limitations that motivate the current work.

Chapter 3 — System Design and Methodology: This chapter describes the overall system architecture, functional and non-functional requirements, system modules, and data flow. It explains how different components such as video search, transcription, visual analysis, summarization, and interaction are integrated within the E-VISTA platform.

Chapter 4 — Implementation Details: This chapter presents the technical implementation of the project. It covers frontend development, backend services, AI model integration, database design, and communication between different modules. Technologies and tools used in the implementation are explained in detail.

Chapter 5 — Results and Analysis: This chapter discusses the results obtained from the developed system. It includes screenshots of the application, summary quality evaluation, transcription performance, and system behavior analysis. The effectiveness of the proposed approach is assessed based on observed outcomes.

Chapter 6 — Conclusion and Future Work: This chapter summarizes the overall contributions and achievements of the project. It also highlights current limitations and

outlines potential future enhancements, such as advanced analytics, additional AI features, and improved scalability.

Chapter 7 — Bibliography: This section lists all research papers, books, standards, and technical documentation consulted throughout the project related to educational video analysis, speech-to-text transcription, multimodal learning, artificial intelligence, and full-stack system development.

Chapter 8 — Appendices: This section includes supplementary material such as the Sustainable Development Goals (SDGs) addressed by the project, self-assessment of the completed work, additional system screenshots, sample outputs, and detailed technical specifications of the E-VISTA platform.

Chapter 2

LITERATURE SURVEY

Sindhu et al. [1] proposed an educational video discovery framework based on advanced latent semantic analysis to improve the relevance of video search results. Their approach captures semantic similarity between user queries and video metadata, enabling more accurate retrieval of topic-related educational content. The study demonstrates that semantic-based search significantly outperforms keyword-based methods, motivating the development of an intelligent video search module. In particular, they show that incorporating semantic relationships between concepts reduces retrieval of off-topic videos and improves the diversity of relevant results.

Javed et al. [2] presented an intelligent parametric cognitive assessment model for e-learning environments that evaluates learner understanding using multiple parameters such as scores, attempts, and behavioral patterns. Their findings show that automated, parameter-driven assessment can reliably estimate student performance and learning progress. This research supports the integration of quiz-based and data-driven evaluation mechanisms in intelligent learning platforms, aligning with the assessment objectives of E-VISTA. The authors further emphasize mapping assessment items to cognitive levels (e.g., Bloom's taxonomy) and aggregating heterogeneous parameters through computational models to obtain a more holistic view of learner proficiency.

Shahzad et al. [3] introduced a multi-agent system for cognitive assessment in e-learning environments, where autonomous agents monitor learner activities, analyze performance, and adapt assessment strategies in real time. Their results indicate that multi-agent architectures improve flexibility and personalization in learner evaluation. This approach closely aligns with E-VISTA's goal of providing adaptive and data-driven learning experiences. By distributing responsibilities across specialized agents (such as monitoring, analysis, and feedback agents), the system can react dynamically to changes in learner behavior and context. The study highlights that such agent-based designs support

scalability and continuous assessment, which are crucial when handling large numbers of learners and high-volume interaction data in modern educational platforms.

Le Scao et al. [4] proposed Mixtral, a mixture-of-experts large language model that routes tokens through specialized expert networks to improve efficiency and model capability. Their architecture demonstrates that modern LLMs can deliver powerful language generation while reducing computational cost. This work motivates the adoption of advanced LLMs in E-VISTA for tasks such as video summarization and question answering. Mixtral's sparse routing mechanism allows only a subset of expert networks to be activated per token, resulting in strong performance comparable to much larger dense models, while maintaining lower inference cost. For a system like E-VISTA, this design is particularly attractive because it enables high-quality language understanding and generation (e.g., summaries, explanations, translations) within limited computational resources.

Chen et al. [5] investigated the use of large language models for automatic question generation from instructional text. Their study shows that LLM-based approaches can generate coherent and pedagogically meaningful questions. These findings directly support the quiz-generation feature of E-VISTA. Beyond factual questions, the authors report that LLMs can be guided—through prompt design or fine-tuning—to produce items targeting different cognitive levels and learning objectives. They also discuss challenges such as controlling difficulty, avoiding ambiguity, and ensuring alignment with curriculum goals, which are critical considerations when integrating automatic question generation into educational systems like E-VISTA.

Giannakos et al. [6] conducted a systematic review of video-based learning, focusing on learner engagement and interaction patterns. Their analysis highlights the importance of structured content and adaptive learning mechanisms in improving educational outcomes. This review supports the integration of intelligent video analysis and abstraction techniques within E-VISTA. The authors synthesize findings from multiple studies on how features such as video length, segmentation, embedded questions, and interactive controls influence attention and retention. They also emphasize the role of learning analytics, including clickstream patterns, in understanding how learners interact with videos. These insights guide the design of E-VISTA's summarization, navigation, and personalization features to align with evidence-based practices in video-based learning.

Chapter 3

System Design & Methodology

3.1 Functional & Non-Functional Requirements

This section outlines the functional and non-functional requirements of the E-VISTA system, which define the core services, operational behavior, and quality constraints of the platform. These requirements serve as the foundation for system design, implementation, and performance evaluation.

3.1.1 Functional Requirements

The system must perform the following functions:

- Allow users to search educational videos across platforms such as YouTube.
- Fetch and display video metadata including title, channel, description, and duration.
- Perform AI-based speech-to-text transcription for selected videos.
- Generate structured summaries from transcribed content using language models.
- Support multilingual translation of transcripts and summaries.
- Extract and analyze visual content such as slides and diagrams.
- Automatically generate quizzes and flashcards from the summarized content.
- Provide an interactive chat interface for query-based learning.
- Maintain user profiles and learning history.

3.1.2 Non-Functional Requirements

In addition to functional behavior, the system should satisfy the following non-functional requirements:

- High transcription accuracy (90 percentage).
- Low response time with real-time progress updates.
- Scalability to support multiple concurrent users.
- High availability and reliability of services.
- Secure handling of user data and requests.
- Platform-independent access through web browsers.

3.2 List of Hardware & Software Requirements

This section specifies the hardware and software resources required for the effective implementation and execution of the E-VISTA system. These requirements ensure smooth AI processing, system stability, and reliable user interaction across the platform.

- **Processor:** Intel Core i5 / AMD Ryzen 5 or higher (required for smooth AI processing and video handling).
- **RAM:** Minimum 8 GB (16 GB recommended for ML models and multiple services).
- **Storage:** Minimum 256 GB SSD (for datasets, videos, and AI model files).
- **GPU (Optional):** NVIDIA GPU with CUDA support (for faster Whisper transcription and ML processing).
- **Network:** Stable high-speed internet connection (for video streaming, API calls, and cloud AI services).

3.2.1 Software Requirements

- **Operating System:** Windows / Linux / macOS
- **Programming Languages:**
  - Python 3.x (AI/ML and backend services)
  - JavaScript (Frontend and integration)

- **Backend Framework:** FastAPI (for RESTful APIs and backend services)
- **Frontend Framework:** React.js with Tailwind CSS (for responsive and enhanced user experience)
- **Speech Recognition Model:** OpenAI Whisper (for audio and video transcription)
- **Large Language Model API:** Gemini API (for summarization, translation, and quiz generation)
- **Database:** SQLite (development) / PostgreSQL (production)
- **Development Tools:** Visual Studio Code, Git
- **API Testing Tools:** Postman / Insomnia
- **Browser:** Google Chrome / Mozilla Firefox

3.3 System Architecture

The E-VISTA system follows a modular three-layer architecture designed to ensure separation of concerns, scalability, and efficient processing of educational video content. The overall workflow and interaction between system components are illustrated in Figure 3.1. The architecture depicts the transformation of raw educational video input into structured learning materials through a sequence of AI-driven processing stages.

As shown in Figure 3.1, the system architecture is organized into interconnected modules that collectively enable intelligent video analysis and content generation. The key components of the architecture are described below:

1. **Educational Video Input & Search Retrieval Module:** This module serves as the entry point of the system, where users provide search queries or select educational videos from external platforms. Relevant videos are retrieved using platform APIs, and metadata such as title, duration, and description is fetched to ensure academic relevance and efficient navigation.
2. **Multi-Modal Analysis Pipeline:** This layer performs comprehensive analysis of the selected video using multiple AI techniques. It includes speech detection and

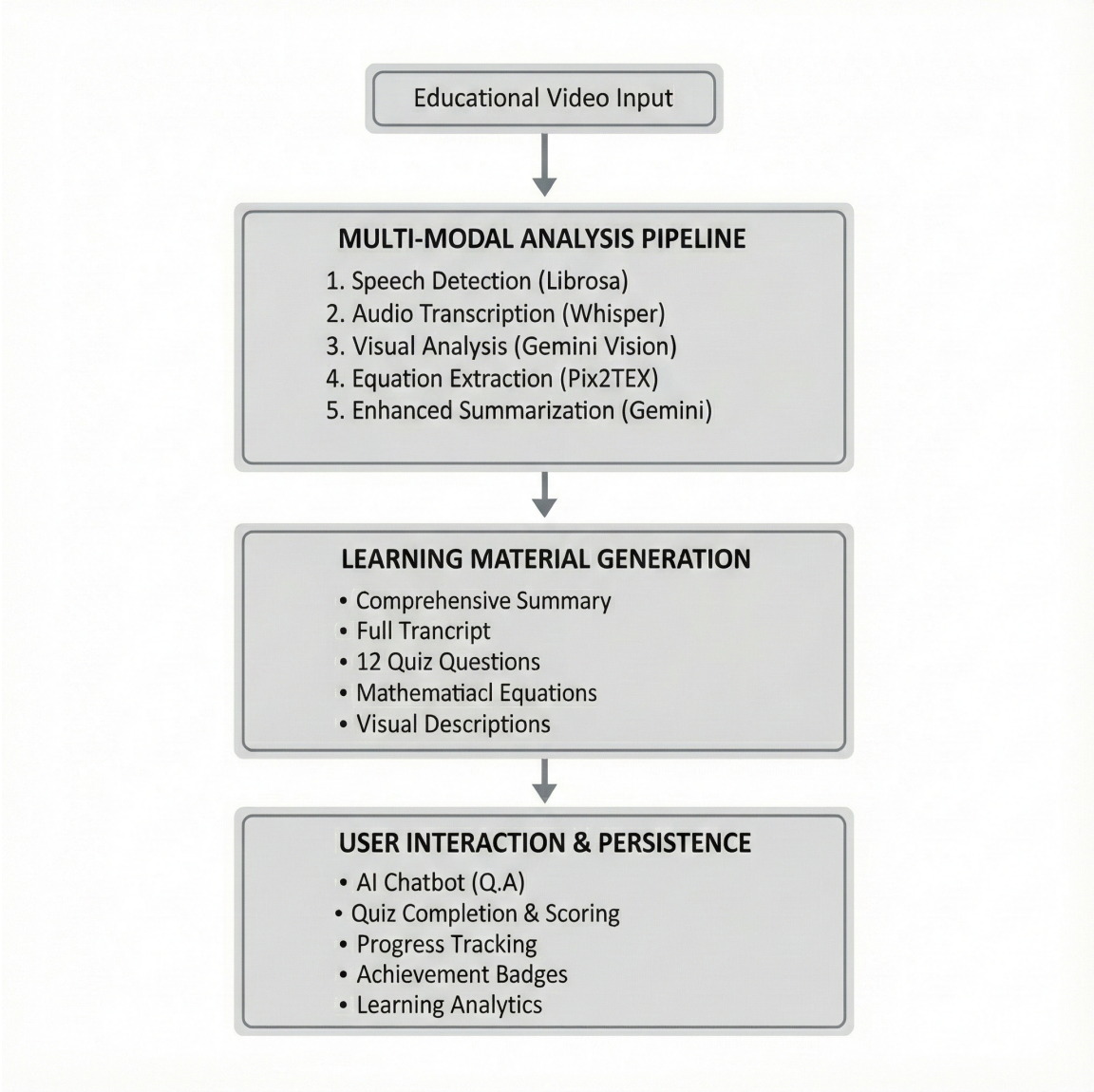

Figure 3.1: Layered architecture of the edu-video insight search, translation and abstraction.

audio transcription using the Whisper model, visual analysis using Gemini Vision, equation extraction from visual content, and enhanced summarization using Gemini-based language models. This pipeline converts unstructured audio-visual data into structured and meaningful information.

3. **Learning Material Generation Module:** Based on the processed outputs of the analysis pipeline, this module generates learning resources such as comprehensive summaries, full transcripts, quiz questions, mathematical equations, and visual descriptions. These materials are structured to support effective learning and quick

concept understanding.

4. **User Interaction & Persistence Layer:** This layer manages user interaction and data persistence. It provides features such as AI-based chatbot support, quiz completion and scoring, progress tracking, achievement badges, and learning analytics. User data and generated content are stored securely to enable personalized and continuous learning experiences.

3.4 Data Flow Diagram

The data flow diagram illustrates the sequential flow of information within the E-VISTA system, explaining how user input is processed and transformed into structured learning outputs. The overall workflow of the system is depicted in Figure 3.2, highlighting the interaction between the user interface, backend services, AI processing modules, and data storage components.

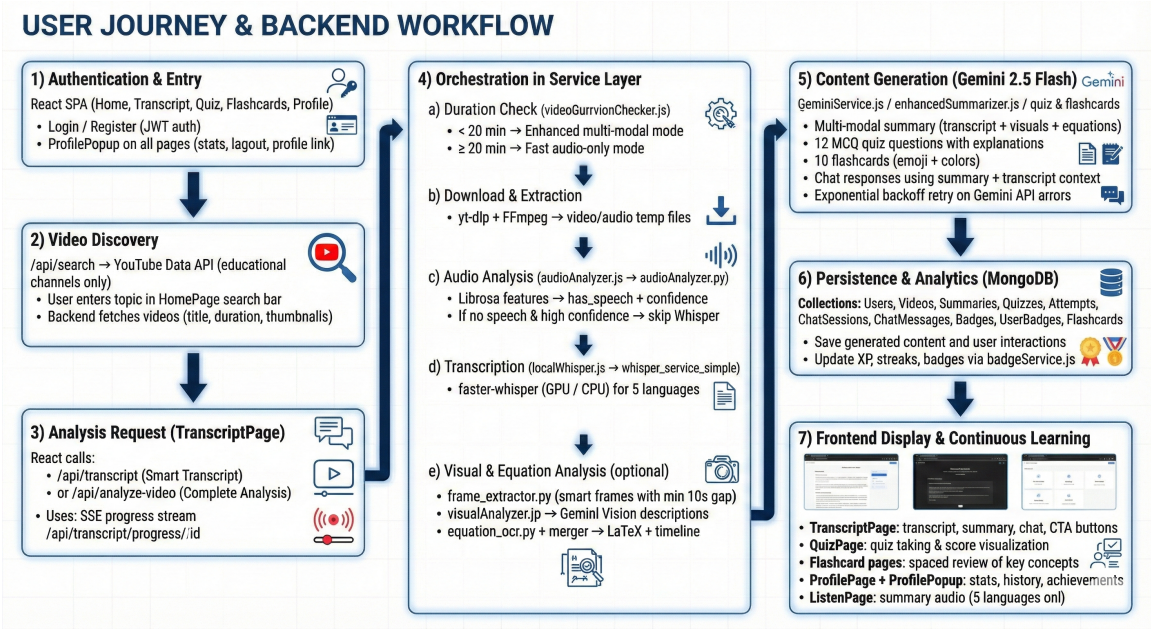

Figure 3.2: Workflow model of edu-video insight search, translation and abstraction.

A typical data flow through the system proceeds through the following stages, as depicted in Figure 3.2:

Step 1: User Input

The process begins with user interaction, where the user provides a search query or selects an educational video through the frontend interface. This input serves as the trigger for initiating the backend processing workflow.

Step 2: Video Retrieval

Based on the user input, the system retrieves relevant video metadata and streaming information from external platforms using platform-specific APIs. This step ensures that only academically relevant videos are selected for further processing.

Step 3: Audio & Visual Extraction

Once a video is selected, the audio stream is extracted and forwarded for speech processing. In parallel, video frames are captured at regular intervals to enable visual analysis of slides, diagrams, and other on-screen content.

Step 4: AI Processing

The extracted audio is processed using the Whisper speech recognition model to generate accurate and time-stamped transcripts. The transcribed content is then analyzed using Gemini-based language models to produce structured summaries from the video.

Step 5: Content Enrichment

In this stage, the summarized content is further enriched through multilingual translation, automatic quiz generation, and flashcard creation. These features enhance learner engagement and support active learning.

Step 6: Storage & Presentation

Finally, all generated outputs, including transcripts, summaries, quizzes, and learning analytics, are securely stored in the database. The processed results are presented to the user through the dashboard interface, enabling progress tracking and continuous learning.

3.5 Algorithms

1. Video Input Processing and Content Extraction Algorithm

This algorithm begins by accepting video or audio input from the user in supported formats. Before processing, the system validates the input based on duration, file size, and media type to ensure compatibility. If the input contains video data, the audio stream is extracted for further processing. The extracted audio is normalized by adjusting parameters such as sampling rate and channel configuration to meet the requirements of the speech recognition model. Relevant metadata including duration, format, and upload timestamp is stored in the database, and the processed audio data is forwarded to the speech recognition module.

2. Speech-to-Text Transcription Algorithm

In this stage, the system receives normalized audio input from the preprocessing layer and detects the available compute device, such as CPU or GPU, to optimize performance. The audio is processed using the **Whisper speech recognition model**, which converts spoken content into text. The generated transcription is segmented into time-stamped sentences or paragraphs to preserve contextual flow. Language detection is performed to support multilingual inputs, and the structured transcript is returned to the backend service for further analysis.

3. AI-Based Content Summarization Algorithm

The summarization algorithm receives the transcript along with optional visual cues from the analysis module. The transcript is preprocessed to remove noise and redundant segments, ensuring clarity and relevance. Based on the required summary type, such as short, detailed, or visual-aware, a dynamic prompt is constructed and sent to the **Gemini Language Model API**. The system applies retry mechanisms and rate-limit handling to ensure stable API interaction. Structured summaries with headings and key highlights are generated and stored in the database.

4. Automatic Quiz Generation and Evaluation Algorithm

This algorithm uses the generated summaries as input to create assessment content. The language model generates multiple-choice questions based on key concepts identified in the summarized content. Answer options are randomized, and correct responses are marked automatically. The quiz is presented to the user through the frontend interface, where responses are captured and evaluated. The system computes the quiz score and updates user learning statistics and quiz history accordingly.

5. User Dashboard and Progress Tracking Algorithm

The progress tracking algorithm aggregates user activities such as generated summaries, quiz attempts, and audio interactions. It retrieves recent transcripts, summaries, and assessment results from the database to analyze user performance. Learning metrics including completion count and accuracy are computed, and achievement badges are updated based on predefined rules. The summarized performance information is then displayed on the user dashboard to support continuous monitoring and personalized learning.

Chapter 4

Implementation Details

Overview

The E-VISTA (AI-Powered Educational Video Analysis Platform) integrates artificial intelligence models, full-stack web technologies, and database-backed storage into a unified learning system. The implementation focuses on transforming educational videos into structured knowledge through transcription, summarization, quizzes, and interactive learning features.

A modular and layered architecture is adopted to ensure scalability, maintainability, and real-time feedback during long-running AI operations such as transcription and summarization.

The major components of the implementation are:

- **AI Processing Module (Python-based):** Handles video analysis, audio extraction, speech-to-text transcription, summarization, translation, quiz generation, and visual content processing.
- **Backend Application Layer (FastAPI-based):** Manages API requests, orchestrates AI workflows, handles authentication, and communicates with the frontend in real time.
- **Frontend User Interface (React-based):** Provides an interactive dashboard for search, progress tracking, results visualization, and learner interaction.
- **Database Layer (SQLite/PostgreSQL):** Stores user data, transcripts, summaries, quizzes, and learning history.

Educational videos are accessed dynamically through supported public platforms using their APIs. AI models such as Whisper and Gemini form the core intelligence of the system and are integrated through well-defined service interfaces.

4.1 AI Processing Implementation

The AI processing layer forms the backbone of the E-VISTA system. It is implemented using Python and is responsible for converting video data into meaningful textual and interactive learning outputs.

4.1.1 Dataset and Input Handling

Unlike offline machine learning systems, E-VISTA operates on live video inputs selected by users. The input data primarily includes video URLs or search results obtained from platforms such as YouTube and Vimeo, audio streams extracted from educational videos, and video frames sampled periodically for visual analysis.

The audio stream is extracted using multimedia processing tools and prepared for speech recognition. Video frames are sampled at fixed intervals to balance accuracy and computational cost during visual analysis.

4.1.2 Speech-to-Text Transcription

Speech-to-text conversion is implemented using the Whisper model, which supports multilingual transcription and long-duration audio. Audio inputs are split into manageable segments to support videos longer than twenty minutes, and each segment is processed independently to ensure system stability.

Time-stamped text segments generated by the model are merged to produce a complete and coherent transcript. Real-time progress updates are sent to the frontend during transcription, enabling better user interaction and transparency.

The system achieves high transcription accuracy (above 90 percent), making it suitable for educational content containing technical terms and structured explanations.

4.1.3 Summary Generation

Once transcription is completed, the transcript is passed to the Gemini-based language model for summarization. The transcript is segmented into coherent sections, and key concepts, definitions, and explanations are identified. Redundant or irrelevant information is removed to improve clarity.

Based on this processed content, a structured summary of approximately five hundred

words or more is generated. The summarization process preserves the educational flow of the original video and ensures conceptual clarity for learners.

4.1.4 Quiz and Flashcard Generation

To support active learning, the system automatically generates assessment material from the summarized content. Important concepts are extracted and used to generate multiple-choice and short-answer questions.

Flashcards are created with concise concept definitions, and all generated outputs are optimized for self-assessment and revision. This automation significantly reduces the need for manual content creation by instructors.

4.1.5 Visual and Mathematical Content Processing

Educational videos often contain slides, diagrams, and mathematical expressions that are essential for understanding technical concepts. Video frames are analyzed using computer vision techniques to detect and extract slide text and diagrams.

Mathematical expressions present in visual content are identified and converted into readable structured formats. This ensures that important visual explanations are not lost during audio-based processing and are preserved as part of the learning output.

4.2 Backend and Frontend Integration

The backend and frontend integration layer ensures seamless communication between AI services, databases, and the user interface. The backend is implemented using FastAPI, which acts as a coordination layer for handling requests, executing AI tasks, and delivering results to the frontend.

4.2.1 Backend Implementation

The backend implementation is responsible for managing system logic and AI orchestration. RESTful APIs are used to handle video search requests, initiate processing workflows, and retrieve results. Asynchronous task execution is employed to manage long-running AI operations such as transcription and summarization without blocking user requests.

Real-time status updates are delivered to the frontend using Server-Sent Events (SSE) or WebSockets, enabling continuous progress monitoring. Security mechanisms are implemented to validate user requests and prevent unauthorized access or misuse of backend resources.

4.2.2 Frontend Implementation

The frontend is implemented using React.js along with modern UI frameworks to provide an interactive and responsive user experience. Users can search for educational videos, view metadata, and initiate AI-based processing directly through the interface.

Real-time progress indicators display the status of transcription and summarization tasks. Generated outputs such as summaries, quizzes, and flashcards are rendered dynamically. User profiles maintain learning progress, recent activity, and performance metrics to support personalized learning.

4.3 System Integration and Workflow

The overall system functions as a continuous learning pipeline, integrating input acquisition, AI processing, and result delivery.

The end-to-end workflow is as follows:

1. Users search for or select an educational video through the frontend.
2. The backend retrieves video metadata and initializes the processing pipeline.
3. Audio and visual data are extracted from the video.
4. The AI engine performs transcription, summarization, translation, and content generation.
5. Generated outputs are stored in the database and streamed to the frontend.
6. Users interact with summaries, quizzes, flashcards, and AI chat features.

The tight integration of AI modules, backend services, frontend visualization, and persistent storage ensures a seamless and responsive learning experience.

Chapter 5

Results and Analysis

This chapter presents the results obtained from the implementation of the E-VISTA system and provides an analysis of its performance and functionality. The outcomes are evaluated based on system behavior, quality of generated content, and user interaction features. Screenshots and observations are used to demonstrate how the proposed system effectively supports intelligent video search, transcription, summarization, and interactive learning.

5.1 Test Procedures and Test Cases

5.1.1 Overview of the Testing Approach

A structured and repeatable testing methodology was followed in a controlled environment to ensure that test results are consistent and reproducible. Testing primarily focused on real-world user workflows such as user registration, video or audio uploads, content summarization, quiz and flashcard generation, AI-based chat interaction, and profile management. Along with functional validation, non-functional characteristics including system latency, stability, and behavior under concurrent usage were also evaluated.

Both automated and manual testing techniques were employed. Automated test scripts were used to capture execution timings, API responses, and processing events, whereas manual testing was carried out to verify user interface behavior, access permissions, and data integrity across the system.

5.1.2 Preconditions

Prior to executing the test cases, the following prerequisites were verified:

- Backend and frontend servers were operational and successfully connected to the database.

- Required test user accounts, including admin and regular users, were created.
- Sample video and audio files were available for upload testing.
- API keys and environment variables were properly configured.
- All necessary software libraries and dependencies were installed.

5.1.3 General Testing Procedure

The following steps were performed for each end-to-end test scenario:

1. The user registers or logs into the system through the frontend.
2. The user uploads a video or audio file, or submits a valid video link.
3. The backend processes the input, generates a transcript and summary, and stores the results in the database.
4. The user creates quizzes or flashcards based on the generated summary.
5. The user attempts quizzes and reviews the evaluation results.
6. The user interacts with the AI chat assistant for additional explanations or clarification.
7. All outputs are validated through the user interface and verified against database records.

Post-condition: Upon successful execution, the system displays the expected outputs such as summaries, quizzes, flashcards, and chat responses, and the database accurately reflects the updated system state.

5.1.4 Detailed Test Cases

Table 5.1: Detailed Functional Test Cases

ID	Test Focus	Description / Steps	Expected Outcome
TC01	User Registration / Login	Register or log in using valid credentials	User is authenticated and redirected to the dashboard
TC02	Video Upload	Upload a supported video or audio file	File is processed and transcript and summary are displayed
TC03	Invalid File Upload	Upload an unsupported file format	Appropriate error message is displayed
TC04	Summarization	Request summary for uploaded media	Summary is generated and shown in the interface
TC05	Quiz Generation	Generate quiz from the summary	Quiz questions are created and displayed
TC06	Quiz Attempt	Answer quiz questions	Score and feedback are shown after submission
TC07	Flashcard Creation	Create flashcard from selected summary text	Flashcard is saved and available in the user profile
TC08	AI Chat Interaction	Ask a question related to content	Context-aware response is generated by the AI assistant
TC09	API Health Check	Access the API health endpoint	JSON response confirming API availability
TC10	Concurrent Uploads	Multiple users upload files simultaneously	System remains stable with no data loss
TC11	Data Consistency	Refresh UI after user actions	Updated and consistent data is displayed
TC12	Permission Handling	Attempt admin-only actions as a regular user	Access is denied with a relevant error message

TC13	Latency Evaluation	Measure processing time for summaries and quizzes	Processing time remains within acceptable limits
TC14	Failure Recovery	Restart backend or frontend during processing	System recovers without loss of user data

5.1.5 Test Data and Automation

- **Sample Media Files:** A collection of over twenty video and audio files of varying duration and content was used for testing.
- **User Accounts:** Multiple admin and regular user accounts were created to validate role-based access control.
- **Automation Suite:** Scripts were developed to automate uploads, trigger summarization, generate quizzes, and collect performance metrics.

5.2 Analysis

5.2.1 System Performance

Summarization and quiz generation typically completed within 5–10 seconds for standard media files. The system maintained stable performance even when handling concurrent uploads. Invalid inputs and unauthorized actions were handled gracefully with clear error messages, and role-based access controls effectively restricted administrative operations.

5.2.2 Consistency and Correctness

The frontend and backend remained synchronized throughout testing, with no instances of stale or conflicting data. All user activities, including uploads, quiz attempts, and flashcard creation, were accurately recorded and reflected in both the database and user interface.

5.2.3 Robustness

The system demonstrated strong resilience during simulated failures. Backend and frontend restarts did not result in data loss, and users were able to resume their activities

without disruption. Concurrent interactions by multiple users were handled reliably.

5.2.4 Observed Trends

Occasional increases in latency were observed during large media uploads or periods of heavy usage; however, the system remained responsive. The overall architecture supports scalability through the addition of backend workers or optimization of media processing components.

5.2.5 Results – Dashboard Screenshots

This section presents the key dashboard screens of the NG-FIREWALL monitoring interface, which serve as the primary visualization layer of the system. These screens provide a comprehensive overview of network activity by representing live traffic flows, detected security events, and system status in an intuitive and user-friendly manner.

Together, the dashboard views illustrate how the system continuously monitors network traffic, identifies potential threats using machine learning models, and highlights suspicious activities in real time. Visual indicators and summary panels enable security analysts to quickly assess the current security posture of the network and prioritize response actions.

In addition, the dashboard exposes detailed model statistics and threat-intelligence summaries, offering insights into detection accuracy, alert frequency, and historical attack patterns. By consolidating traffic monitoring, threat analysis, and intelligence reporting into a unified interface, the dashboard enhances situational awareness and supports efficient decision-making for security analysts.

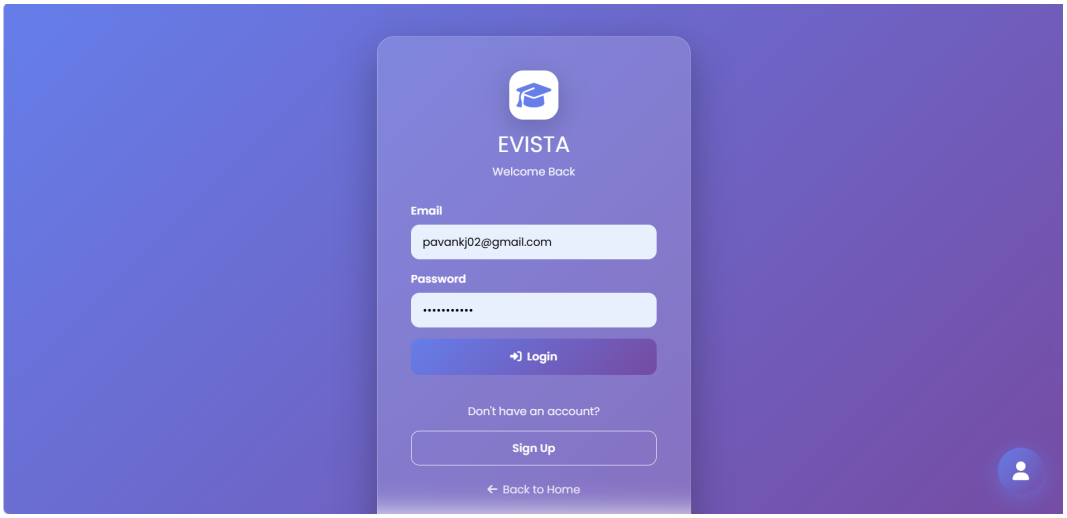

Figure 5.1: E-VISTA Login Page – User authentication interface to securely access the platform. This page ensures authorized access through credential verification and serves as the entry point to the system by initializing user sessions securely. Authentication mechanisms prevent unauthorized access and protect user data.

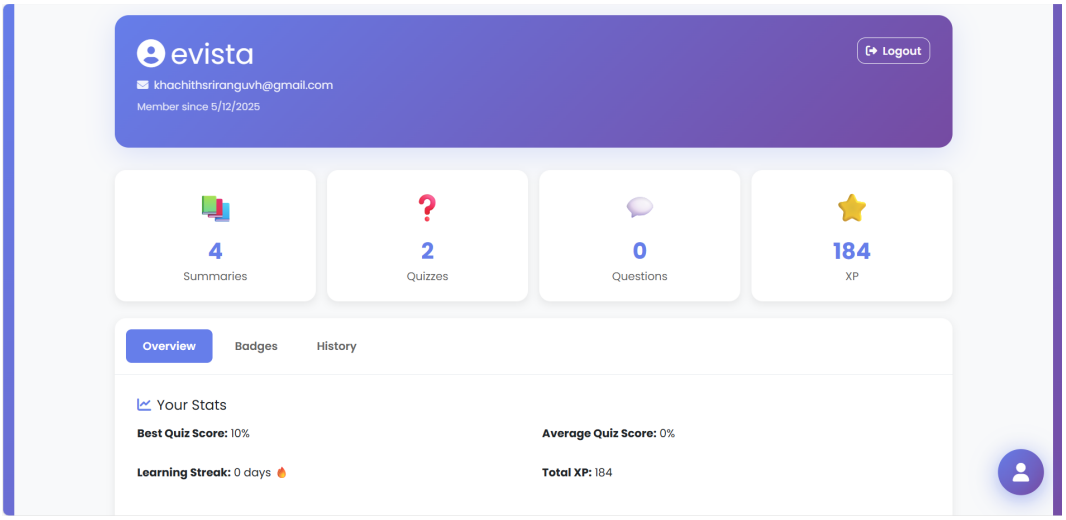

Figure 5.2: Dashboard Page – Central control panel displaying navigation cards for search, summaries, quizzes, and flashcards. It provides a unified view of system features and enables smooth navigation between learning modules.

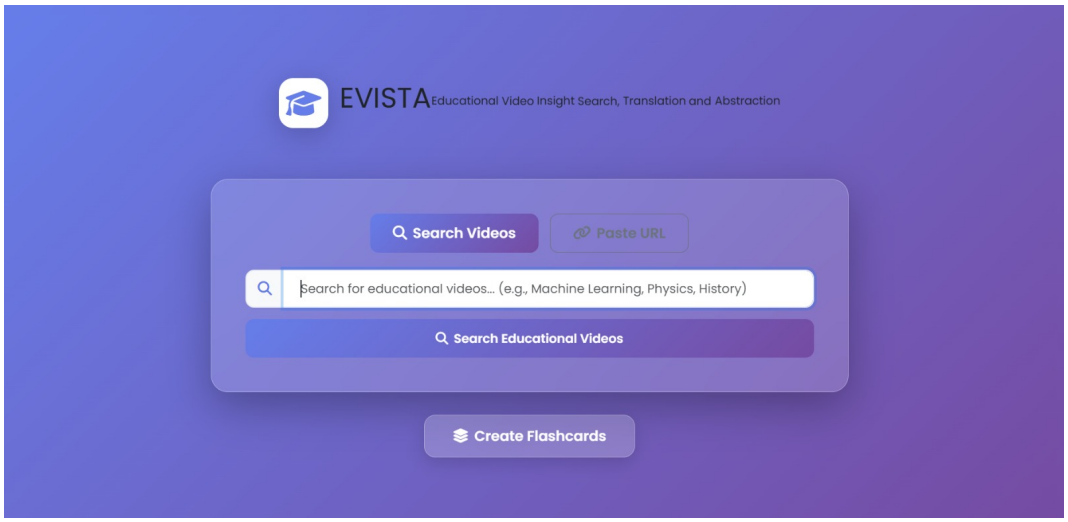

Figure 5.3: Intelligent Video Search – Search results generated based on user input keywords and queries. This interface allows users to explore relevant educational videos and select content aligned with their learning objectives. Filtering and ranking techniques improve the relevance of retrieved results and help users quickly identify appropriate learning resources.

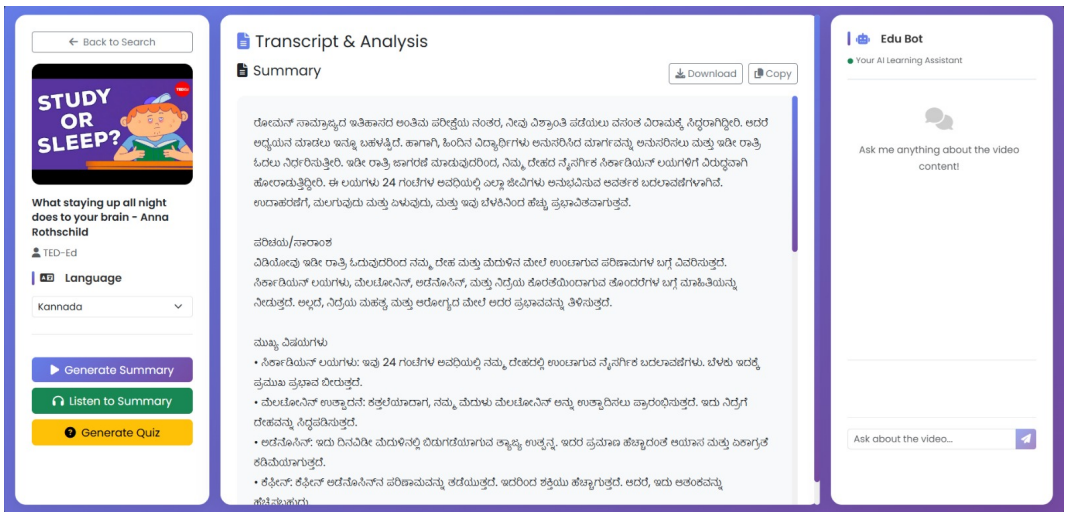

Figure 5.4: Transcript Progress View – Real-time transcription and highlighting of video speech for better comprehension. The synchronized display enables users to track spoken content accurately as the video progresses.

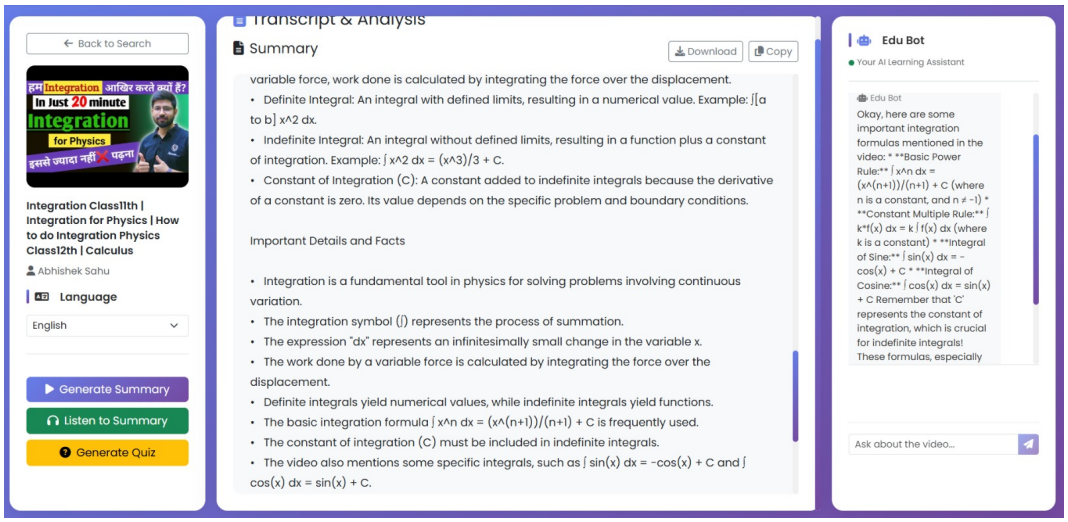

Figure 5.5: Structured Summary Generation – Condensed text summaries automatically produced from the video content. The summaries organize key concepts and explanations into readable sections for faster understanding. This structured abstraction reduces cognitive load and supports efficient revision of complex topics.

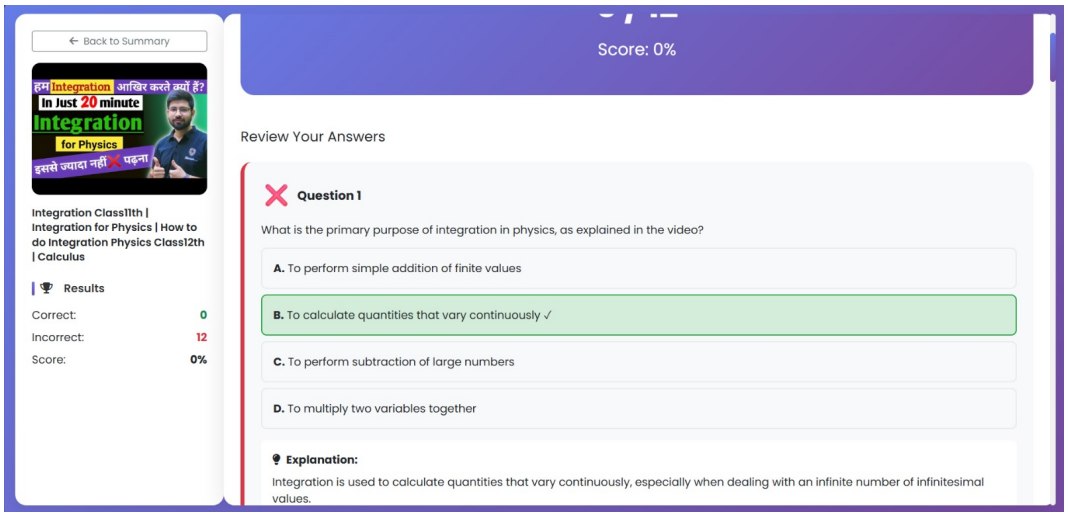

Figure 5.6: Quiz Generation – Auto-created quiz questions to test understanding of summarized concepts. This feature promotes active learning by reinforcing important topics through assessment-based interaction.

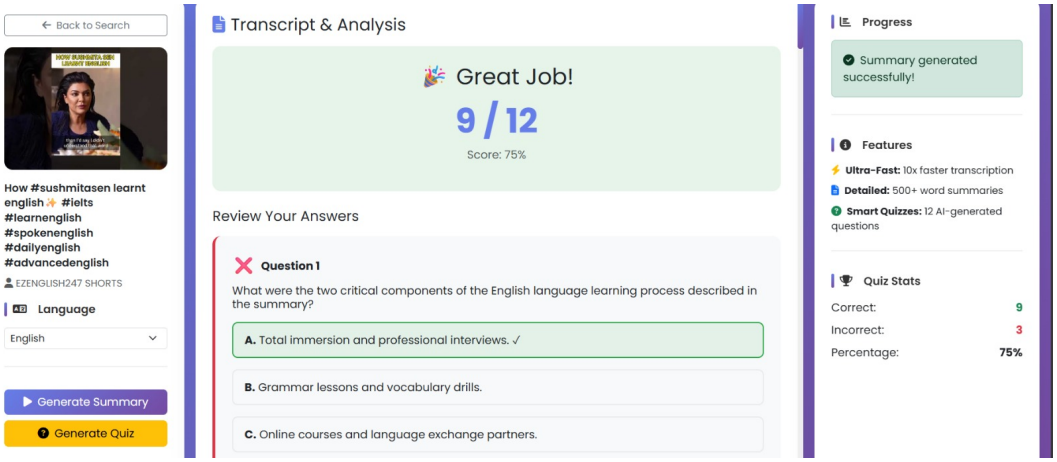

Figure 5.7: Quiz Results and Correct Answers – Displays user responses along with correct answer validation and feedback. It helps learners evaluate performance and understand mistakes for improved learning outcomes. Performance insights assist users in identifying weak areas and improving future learning strategies.

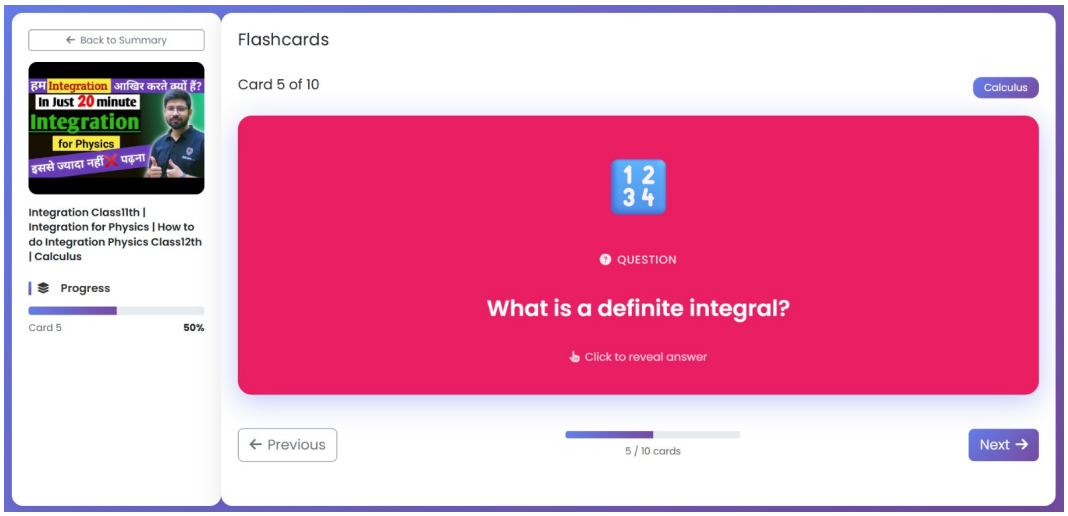

Figure 5.8: Flashcards Interface – Key learning points converted into flashcards for quick review and revision. This supports repeated practice and enhances long-term retention of important concepts.

Chapter 6

Conclusion and Future Work

6.1 Summary of Achievements

This mini project successfully designed and implemented E-VISTA, an AI-powered educational video analysis platform that transforms unstructured video-based learning content into structured, accessible, and interactive educational resources. By integrating modern artificial intelligence techniques with a full-stack web architecture, the project addresses key limitations of traditional video-based learning.

The major achievements of the project include:

- **Intelligent Educational Video Retrieval:** Implemented a unified video search mechanism that retrieves relevant educational content from platforms such as YouTube. The system filters results using curated educational channels, ensuring that users receive academically relevant and high-quality content.

- **AI-Powered Speech-to-Text Transcription:** Integrated the Whisper speech recognition model to convert long educational videos into accurate text transcripts. The system supports videos longer than 20 minutes and provides real-time progress updates, achieving transcription accuracy greater than 90 percent.

- **Structured Summary Generation:** Developed an automated summarization pipeline using Gemini-based large language models to generate concise yet comprehensive summaries. This significantly reduces the time required for learners to grasp key concepts from lengthy videos.

- **Multimodal Content Understanding:** Extended analysis beyond audio by incorporating visual content processing, enabling extraction of information from slides, diagrams, and mathematical expressions commonly used in educational videos.

- **Interactive Learning Support:** Automatically generated quizzes, flashcards, and AI-driven question–answer interactions from summarized content, transforming passive video consumption into an active learning experience.

- **Multilingual Accessibility:** Enabled multilingual translation of transcripts and summaries, supporting languages such as English, Hindi, Tamil, Telugu, and Kannada, thereby improving accessibility for a wider audience.

- **User-Centric Dashboard and Learning History:** Designed a responsive web-based dashboard that visualizes transcription progress, summaries, assessments, and user learning history, supporting personalized and self-paced learning.

6.2 Key Contributions

The primary contributions of this project can be summarized as follows:

- **End-to-End Educational Video Analysis Platform:** Developed a complete proof-of-concept system that covers video retrieval, transcription, summarization, assessment generation, and result visualization within a single unified platform.

- **Multimodal AI Integration:** Demonstrated effective integration of speech recognition, natural language processing, and visual analysis to capture both audio and visual educational information.

- **Automation of Learning Content Creation:** Eliminated the need for manual creation of summaries and quizzes by instructors through AI-based automation, improving efficiency and scalability.

- **Learner-Focused System Design:** Designed the platform with a focus on learner usability, real-time feedback, and personalization, making advanced AI capabilities accessible without requiring technical expertise.

- **Modular and Extensible Architecture:** Adopted a layered system design that separates frontend, backend, AI processing, and data storage, enabling future enhancements and easier maintenance.

6.3 Future Work

Several enhancements can further improve the robustness, scalability, and educational impact of the E-VISTA platform.

Enhanced Personalization and Adaptive Learning

- Introduce adaptive learning paths based on user performance and preferences.
- Recommend videos, summaries, and quizzes tailored to individual learners.
- Track learning outcomes more deeply using advanced analytics.

Advanced AI Models and Processing

- Explore transformer-based and multimodal foundation models for deeper semantic understanding of educational content.
- Integrate domain-specific fine-tuned models for subjects such as mathematics, engineering, and medicine.
- Improve visual understanding to better extract diagrams, charts, and handwritten content.

Privacy, Security, and Compliance

- Implement stronger privacy-preserving mechanisms for handling user data and transcripts.
- Provide on-device or local inference options for sensitive educational environments.
- Align data handling and storage with educational data protection regulations.

Bibliography

[1] B. Sindhu, A. Bhaskar, G. Yugesh, S. Reshma, and B. Rohit, "Enhancing Educational Video Discovery Using Advanced Latent Semantic Analysis," Procedia Computer Science, Elsevier, 2025.

[2] M. S. Javed, M. Aslam, and S. K. Khurshid, "An Intelligent Model for Parametric Cognitive Assessment of E-Learning-Based Students," Procedia Computer Science, Elsevier, 2025.

[3] R. Shahzad, M. Aslam, S. Al-Otaibi, M. S. Javed, A. R. Khan, S. A. Bahaj, and T. Saba, "Multi-Agent System for Students' Cognitive Assessment in E-Learning Environment," Journal of Intelligent Fuzzy Systems, 2024.

[4] A. Radford et al., "Robust Speech Recognition via Large-Scale Weak Supervision," OpenAI Whisper Technical Report, 2022.

[5] T. Le Scao et al., "Mixtral: A Mixture of Experts Language Model," Hugging Face Research, 2024.

[6] X. Chen et al., "Large Language Models for Automatic Question Generation," in Proceedings of the Association for Computational Linguistics (ACL), 2022.

[7] M. Giannakos, J. Krogstie, and D. Sampson, "Video-Based Learning: A Systematic Review," Educational Technology Society, 2023.

[8] OpenAI, "Whisper: Speech-to-Text API Documentation," 2022.

[9] Google Research, "Gemini: Multimodal Large Language Models," Google GenAI Documentation, 2024.

[10] CTranslate2 Authors, "Faster-Whisper: Fast and Accurate Speech Recognition," GitHub Repository, 2023.

Appendices

Appendix A

Sustainable Development Goals Addressed

#	**SDG**	**Level**
1	No Poverty	1
2	Zero Hunger	1
3	Good Health and Well-being	1
4	Quality Education	3
5	Gender Equality	2
6	Clean Water and Sanitation	1
7	Affordable and Clean Energy	1
8	Decent Work and Economic Growth	2
9	Industry, Innovation and Infrastructure	3
10	Reduced Inequalities	2
11	Sustainable Cities and Communities	1
12	Responsible Consumption and Production	1
13	Climate Action	1
14	Life Below Water	1
15	Life on Land	1
16	Peace, Justice and Strong Institutions	1
17	Partnerships for the Goals	2

Levels: Poor = 1, Good = 2, Excellent = 3

Appendix B

Self-Assessment of the Project

#	**PO and PSO**	**Contribution from the Project**	**Level**
1	Engineering Knowledge	Applied concepts of artificial intelligence, machine learning, databases, and software engineering in system development	3
2	Problem Analysis	Analyzed limitations of passive video learning and identified needs for automation and personalization	3
3	Design/Development of Solutions	Designed and implemented an AI-driven platform for video transcription, summarization, and assessment	3
4	Conduct Investigation of Complex Problems	Evaluated performance of speech recognition, summarization, and question generation models	2
5	Modern Tool Usage	Used cloud APIs, AI models, web frameworks, databases, and visualization tools	3
6	Engineer and Society	Contributed to improving accessibility and effectiveness of digital education	2
7	Ethics	Ensured responsible use of educational content and user data privacy	2
8	Individual and Team Work	Collaborated in project planning, development, and documentation	3
9	Communication	Prepared technical reports, presentations, and user-friendly interface	3
10	Project Management and Finance	Planned project milestones, time management, and efficient resource usage	2
11	Life-long Learning	Explored emerging AI technologies and tools for educational applications	3

	PSO	Description	Level
1	PSO1	Designed a computer-based intelligent system using AI and backend integration	3
2	PSO2	Developed a complete software solution using programming languages and frameworks	3
3	PSO3	Implemented an internet-based application using cloud services and APIs	2

Levels: Poor = 1, Good = 2, Excellent = 3

Appendix C

Data Sheet of Component 1

Software Components Used in the Project

- **AI Model APIs:** Used for speech recognition, text summarization, translation, and question generation.
- **Backend Server:** Developed using a web framework to handle API requests, processing logic, and database operations.
- **Database:** Used for storing transcripts, summarized content, user queries, and assessment data.

Appendix D

Data Sheet of Component 2

Hardware and Cloud Components

- **Cloud Platform:** Used for hosting backend services and AI processing.
- **Client System:** Standard desktop or laptop with web browser support for accessing the application.
- **Internet Connectivity:** Required for accessing educational videos, cloud APIs, and platform services.