VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

An Autonomous Institute Affiliated to University of Mumbai

Department of Computer Engineering

Project Report on

SmartLearn: Intelligent Learning Platform

In partial fulfillment of the Fourth Year, Bachelor of Engineering (B.E.) Degree in Computer Engineering at the University of Mumbai Academic Year 2024-25

Submitted by

Shrinivas Ghumare (D17 - A, Roll no - 19)
Pratik Wagharalkar (D17 - A, Roll no - 67)
Shubham Chelani (D17 - A, Roll no - 04)
Mayank Wadhwani (D17 - B, Roll no - 66)

Project Mentor

Prof. Mrs. Prerna Solanke

(2024-25)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY
An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering

Certificate

This is to certify that ***Shrinivas Ghumare (D17A - 19), Pratik Wagharalkar (D17A - 67), Shubham Chelani (D17A - 04), Mayank Wadhwani (D17B - 66)*** of Fourth Year Computer Engineering studying under the University of Mumbai have satisfactorily completed the project on "***SmartLearn: Intelligent Learning Platform***" as a part of their coursework of PROJECT-II for Semester-VIII under the guidance of their mentor ***Prof. Mrs. Prerna Solanke*** in the year 2024-25.

This thesis/dissertation/project report entitled ***SmartLearn: Intelligent Learning Platform*** by ***Shrinivas Ghumare (D17A - 19), Pratik Wagharalkar (D17A - 67), Shubham Chelani (D17A - 04), Mayank Wadhwani (D17B - 66)*** is approved for the degree of ***B.E. Computer Engineering***.

Programme Outcomes	Grade
PO1,PO2,PO3,PO4,PO5,PO6,PO7, PO8, PO9, PO10, PO11, PO12 PSO1, PSO2	

Date: 28/04/2025
Project Guide: Prof. Mrs. Prerna Solanke

Project Report Approval
For
B. E (Computer Engineering)

This thesis/dissertation/project report entitled ***SmartLearn: Intelligent Learning Platform*** by ***Shrinivas Ghumare (D17A - 19), Pratik Wagharalkar (D17A - 67), Shubham Chelani (D17A - 04), Mayank Wadhwani (D17B - 66)*** is approved for the degree of ***B.E. Computer Engineering***.

Internal Examiner

External Examiner

Head of the Department

Principal

Date: 28/04/2025
Place: Chembur, Mumbai

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

--	--
Shrinivas Ghumare (D17A - 19)	Mayank Wadhwani (D17B - 66)
--	--
Shubham Chelani (D17A - 04)	Pratik Wagharalkar (D17A - 67)

Date: 28/04/2025
Place: Chembur, Mumbai

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Prof. Mrs. Prerna Solanke** for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to the Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair** , for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement at several times.

Computer Engineering Department

COURSE OUTCOMES FOR B.E PROJECT

Learners will be to,

Course Outcome	Description of the Course Outcome
CO 1	Able to apply the relevant engineering concepts, knowledge and skills towards the project.
CO2	Able to identify, formulate and interpret the various relevant research papers and to determine the problem.
CO 3	Able to apply the engineering concepts towards designing solutions for the problem.
CO 4	Able to interpret the data and datasets to be utilized.
CO 5	Able to create, select and apply appropriate technologies, techniques, resources and tools for the project.
CO 6	Able to apply ethical, professional policies and principles towards societal, environmental, safety and cultural benefit.
CO 7	Able to function effectively as an individual, and as a member of a team, allocating roles with clear lines of responsibility and accountability.
CO 8	Able to write effective reports, design documents and make effective presentations.
CO 9	Able to apply engineering and management principles to the project as a team member.
CO 10	Able to apply the project domain knowledge to sharpen one's competency.
CO 11	Able to develop a professional, presentational, balanced and structured approach towards project development.
CO 12	Able to adopt skills, languages, environment and platforms for creating innovative solutions for the project.

Index

Chapter No	Title	Page No.
1	**Introduction**	10
1.1	Introduction to the project	10
1.2	Motivation for the project	10
1.3	Problem Definition	11
1.4	Existing Systems	11
1.5	Lacuna of the Existing Systems	12
1.6	Relevance of the Project	12
2	**Literature Survey**	13
A	Brief overview of Literature Survey	13
2.1	Research Papers Abstract of the research paper Inference drawn from the paper	13
2.2	Patent Search	14
2.3	Inference Drawn	19
2.4	Comparison with the Existing Systems	20
3.	**Requirement Gathering for the Proposed System**	21
3.1	Introduction to Requirement Gathering	21
3.2	Functional Requirements	21
3.3	Non-Functional Requirements	22
3.4	Hardware, Software, Technology and tools utilised	24
3.5	Constraints	25
4.	**Proposed Design**	26
4.1	Block diagram of the system	26
4.2	Modular design of the system	27
4.3	System Design	29

4.4	Project Scheduling & Tracking using Timeline / Gantt Chart	31
5.	**Implementation of the Proposed System**	32
5.1	Methodology employed for development	32
5.2	Algorithms and flowcharts for the respective modules developed	33
5.3	Datasets source and utilization	37
6.	**Testing of the Proposed System**	38
6.1	Introduction to testing	38
6.2	Types of tests Considered	38
6.3	Various test case scenarios considered	39
7.	**Results and Discussions**	41
7.1	Screenshots of User Interface (UI) for the respective module	41
7.2	Performance Evaluation measures	44
7.3	Input Parameters / Features considered	45
7.4	Inference drawn	46
8.	**Conclusion**	47
8.1	Limitations	47
8.2	Conclusion	47
8.3	Future Scope	48
	References	49
	Appendix	51
1	**Paper I**	51
a	Paper I	51
b	Plagiarism Report of Paper I	58
c	Project review sheet	59

LIST OF FIGURES

Figure No	Heading	Page No
4.1	Block diagram of the System	26
4.2	Modular Diagram	27
4.3	Detailed Design of the System	29
4.4	Gantt chart	31
5.2.1	Flowchart of Quiz Generation module	34
5.2.2	Flowchart of Lectures And Notes Module	35
5.2.3	Flowchart of Lectures (Student Side)	36
7.1.1	Home Page	41
7.1.2	Login Page	41
7.1.3	Quiz Management Dashboard	42
7.1.4	Classroom Announcement Interface	42
7.1.5	Assignment Section	43
7.1.6	Quiz Interface	43

Abstract

In the rapidly evolving landscape of digital education, the demand for interactive and accessible learning platforms is greater than ever. This study introduces a comprehensive Smart Classroom system designed to enhance both teaching effectiveness and student engagement through an integrated digital environment. The platform features a dual-login system: one for students to access customized learning resources—including lecture videos, assignments, assessments, and attendance—organized by academic year, branch, semester, and subject; and another for teachers to manage educational content, monitor student performance, and conduct virtual classes.

A standout feature of the system is the "air canvas," a machine learning-powered tool that enables instructors to use hand gestures as a virtual pen during live lectures. This functionality, developed using Python libraries and computer vision techniques, transforms traditional teaching by allowing real-time drawing and annotation without physical contact with the screen. By bridging physical and digital interaction, the air canvas enhances instructional clarity and student comprehension.

Built with scalability and user-centric design, the system supports a wide range of engineering disciplines and academic levels. It ensures a seamless user experience while maintaining robust performance under varying institutional requirements. This paper discusses the entire development lifecycle, including system architecture, technology stack, and implementation challenges, with an emphasis on practical deployment and classroom impact.

The results underscore the potential of smart classroom technologies to revolutionize educational delivery. By integrating real-time interaction tools and structured content management, the proposed system significantly improves the quality and accessibility of education, paving the way for a more connected and effective digital learning ecosystem.

Chapter 1: Introduction

1.1. Introduction:

The rapid advancement of technology has reshaped many industries, and education is no exception. Traditional classroom methods are increasingly being enhanced by digital platforms that offer greater flexibility, accessibility, and interactivity. In this context, the Smart Classroom system proposed in this project aims to modernize engineering education by providing a dynamic and efficient digital learning environment.The system features two tailored interfaces: one for students, allowing access to lecture videos, assignments, assessments, and attendance records—categorized by year, branch, semester, and subject—and another for teachers (admins), enabling content management, student performance tracking, and live online lecture delivery.A standout feature is the "air canvas", a real-time hand-tracking tool developed using Python and machine learning, which allows teachers to draw or write in mid-air as if using a virtual pen. This enhances lecture interactivity, replicating the traditional whiteboard experience in a digital format.Designed for scalability and diverse academic needs, the platform supports both in-person and remote learning models, ensuring seamless access to educational content across disciplines.

1.2. Motivation:

Early detection of critical diseases such as lung cancer and Parkinson's disease plays a vital role in improving treatment outcomes and patient survival rates. However, conventional diagnostic methods, including manual CT scan analysis and clinical assessments, are often time-consuming, expensive, and inaccessible to a large section of the population, particularly in remote areas. Delayed diagnosis can lead to disease progression, reducing the effectiveness of medical interventions. With advancements in artificial intelligence and machine learning, there is a growing opportunity to develop automated solutions that enhance early disease detection and streamline healthcare accessibility. This project aims to leverage deep learning for lung cancer detection through CT scan image analysis and machine learning techniques for Parkinson's disease detection using speech analysis. By providing an AI-driven system for quick and accurate diagnosis, this approach minimizes dependency on specialized healthcare professionals and facilitates timely medical attention.

1.3. Problem Definition:

Traditional classroom environments, especially in technical fields like engineering, struggle to meet the growing demands for flexibility, interactivity, and personalized learning. As education shifts toward hybrid and remote models, most existing digital platforms fall short—lacking real-time engagement tools, efficient content management, and features that support hands-on or visual learning. This creates a disconnect between teachers and students, resulting in reduced engagement and poor conceptual understanding. Additionally, educators face difficulties in managing diverse academic materials and delivering interactive lessons without advanced tools like real-time annotation or gesture-based drawing. To address these issues, this project proposes a Smart Classroom system featuring dual interfaces for students and teachers, structured content organization, and a machine learning-powered “air canvas” tool that allows instructors to draw or write in real time using hand gestures. This innovative solution aims to create a more interactive, scalable, and effective digital learning environment that bridges the gap between traditional teaching methods and modern educational needs.

1.4. Existing Systems:

Existing digital education platforms often fall short in addressing the practical needs of technical education, particularly in fields like engineering where visual aids, real-time interaction, and hands-on learning are essential. These systems typically lack integrated tools for live annotation, content customization, and dynamic engagement, resulting in a passive learning experience. Teachers face challenges in managing large volumes of academic content and delivering interactive lectures across various semesters and disciplines. Moreover, current platforms do not support intuitive features like gesture-based writing or drawing, which can replicate the whiteboard experience in an online setting. While some tools offer isolated functionalities, they lack a unified, scalable solution tailored for both in-person and remote learning. Therefore, there is a need for a comprehensive Smart Classroom system that integrates real-time teaching tools, structured content management, and a machine learning-based “air canvas” to enhance engagement, accessibility, and the overall effectiveness of digital education.

1.5. Lacuna of the Existing System:

- Existing digital education platforms primarily focus on content delivery, offering limited support for real-time interaction and active student engagement.
- Most platforms do not provide integrated tools for gesture-based drawing or real-time annotation, restricting interactive teaching methods essential in technical education.
- Learning management systems often follow a generalized structure, lacking the ability to tailor content based on academic year, branch, semester, and subject, leading to poor content organization and access.
- There is limited support for hybrid learning environments that combine both in-person and remote education seamlessly.
- Teachers face difficulties in efficiently managing and updating large volumes of educational content due to the absence of intelligent and scalable content management systems.
- Current systems do not offer a unified, user-friendly platform that combines live lectures, performance tracking, and smart interactive tools in a single solution.

1.6. Relevance of the Project:

The rapid evolution of digital technology, coupled with the growing demand for remote and flexible education, has made the integration of smart learning environments more crucial than ever. This project leverages emerging technologies such as machine learning and computer vision to enhance classroom interactivity and streamline content management. By introducing features like the gesture-based “air canvas” and a dual-interface platform for students and teachers, the system transforms traditional education into an engaging, efficient, and accessible experience. It caters specifically to the diverse and practical needs of engineering education, where visual aids and hands-on learning are essential. The Smart Classroom not only bridges the gap between conventional and digital teaching methods but also supports hybrid learning models, ensuring scalability, adaptability, and relevance in today’s tech-driven academic landscape.

Chapter 2: Literature Survey

A. Overview of literature survey:

The papers discussed focus on the development of smart classrooms, exploring machine learning and computer vision techniques to enhance education. They highlight the integration of AI-driven tools like gesture recognition, real-time annotation, and adaptive learning systems to create more interactive and personalized learning environments. The research emphasizes the optimization of these technologies, including features like the "air canvas" (gesture-based drawing) and dynamic content management, to improve both in-person and remote learning. The key takeaway is that AI can significantly enhance classroom interactivity, content delivery, and student engagement, offering a scalable solution for modern education.

2.1. Research Papers :

a. Abstract of the Research Paper

The impact of new technology on education has attracted the attention of many scholars, with the focus increasingly shifting toward smart classrooms. Schools are promoting educational digitization by implementing smart classroom systems, which integrate classroom teaching (CT), teacher-student activities, and online education tools. This project experiments with the design and application of a smart classroom model that uses deep learning for intelligent classroom teaching. The system features two logins (for students and teachers), where teachers can upload lecture videos, assignments, and attendance reports. A notable feature is the **gesture-controlled pen**, implemented through machine learning, which allows teachers to use their hand as a virtual pen for online lectures. Through experimental analysis and surveys, this study compares the learning outcomes of experimental and control classes in a computer science course.

b. Inference Drawn from the Paper

Machine learning models significantly improve the accuracy and speed of lung cancer and Parkinson’s disease detection.

- AI-driven early detection enables timely interventions, reducing mortality rates.
- The use of deep learning enhances diagnostic accuracy in medical imaging analysis.
- Web-based platforms facilitate widespread accessibility to healthcare solutions.
- Voice-based biomarker analysis is a promising technique for early Parkinson’s detection.

2.2 Patent Search

1. **The Design and Application of Smart Classroom Teaching Mode in Higher Vocational Education Based on Deep Learning.**

Abstract

The impact of new technology on education has attracted the attention of many scholars, with the focus increasingly shifting toward smart classrooms. Schools are promoting educational digitization by implementing smart classroom systems, which integrate classroom teaching (CT), teacher-student activities, and online education tools. This project experiments with the design and application of a smart classroom model that uses deep learning for intelligent classroom teaching. The system features two logins (for students and teachers), where teachers can upload lecture videos, assignments, and attendance reports. A notable feature is the **gesture-controlled pen**, implemented through machine learning, which allows teachers to use their hand as a virtual pen for online lectures. Through experimental analysis and surveys, this study compares the learning outcomes of experimental and control classes in a computer science course.

Inferences

1. Better Learning Outcomes: The experimental group using the smart classroom system with features like the gesture-controlled pen showed improved performance compared to the traditional method. This suggests that integrating interactive technology can enhance learning efficiency.
2. Student Preference: The survey indicates that students prefer the smart classroom model, especially its interactive tools, showing that modern educational technologies align well with student expectations.
3. Technological Integration: By leveraging machine learning algorithms like K-means and MapReduce, the smart classroom can analyze and adapt to student behaviors, offering personalized educational experiences and improving overall classroom management.

2. Construction of Teaching Behaviors Analysis Model for Smart Classroom

Abstract

A Smart classroom is an intelligent and efficient classroom based on the “Internet +” and the new generation information technologies such as big data and cloud computing. The teaching process of smart classrooms contains a large amount of teaching behavior data with the value of research and application. Based on the review of smart classroom teaching behaviors analysis and evaluation, this paper first compared classical classroom teaching behaviors analysis methods at home and abroad, and then proposed the construction principles and process, and designed the smart classroom teaching behaviors analysis model (SCTBA) to provide research foundation for the analysis and evaluation of teaching behaviors on smart classroom.

Inferences

1. Smart Classroom Technology: The paper emphasizes the role of modern technologies such as big data and cloud computing in making classrooms more intelligent and efficient. This is in line with your project's use of machine learning and smart tools like the gesture-controlled pen.
2. Teaching Behavior Data: The research highlights the importance of analyzing teaching behaviors, which is relevant to your project as it can help evaluate how effectively teachers interact with students in a smart classroom environment.
3. Model Development: The paper's focus on building a behavior analysis model (SCTBA) aligns with the need to assess and improve teaching practices using data, which could be useful for implementing continuous improvements in your smart classroom system.

3. Study on Hand Gesture Recognition by using Machine Learning

Abstract:

Artificial Intelligence [AI] and Machine Learning [ML] play a vital role in the healthcare industries and other such scientific applications. In continuation with Artificial Intelligence, so many researchers are attracted towards the Hand Gesture Recognition (HGR). HGR is the simple method used to interact with machines. Hand gesture recognition is performed by using machine learning algorithms. It consists of two approaches, they are appearance-based approach and model-based approach. In an appearance-based method, the hand image is reconstructed by using image properties. But in model-based methods, different models such as volumetric, geometric, etc., are used to reconstruct the image. The complications in machine learning algorithms and

processing time are remaining as the major challenge in gesture recognition. The performance of the HGR system is evaluated based on its accuracy. This paper presents a detailed study on various methods and algorithms used in gesture recognition.

Inferences:

1. Hand Gesture Recognition (HGR): This paper provides a comprehensive study on the methods and machine learning algorithms used in gesture recognition, directly linking to the feature in your project where the teacher uses hand gestures to interact with the classroom system.
2. Challenges in Gesture Recognition: It mentions the challenges in processing time and machine learning algorithms, which could also be relevant to your project, especially if you're exploring ways to enhance the performance and accuracy of gesture recognition.
3. Application of AI and ML: The role of AI and ML in HGR aligns with your use of machine learning to enable real-time hand gesture control in your smart classroom system, further supporting your implementation approach.

4. Smart Classroom Application

Abstract

Smartphones have become essential tools in education, offering students a convenient way to store lecture notes, access educational materials, and serve as digital encyclopedias. Despite their benefits, many educational institutions ban mobile phones during classes due to misuse, such as texting or gaming. This project aims to manage smartphone usage in schools by developing a mobile application that allows institutional authorities to control which apps students can access while on campus, ensuring they are focused on learning. Additionally, the app automates attendance tracking, allowing students to mark their attendance digitally, saving time and reducing the chances of proxy attendance. By ensuring smartphones are used solely for academic purposes, thousands of lecture hours can be saved each year, enhancing the overall efficiency of the educational process.

Inference:

1. Controlled Smartphone Usage: The app ensures that smartphones are used strictly for educational purposes during class time, addressing the common issue of distraction from non-academic activities.
2. Time-Saving Attendance System: Automating attendance through the mobile app significantly reduces the time spent on manual attendance-taking and eliminates proxy attendance, making classes more efficient.
3. Institutional Oversight: Authorities can manage and restrict which applications are accessible to students, providing greater control over the learning environment while leveraging the educational potential of smartphones.

5. Impact of Smart Classroom: A Study

Abstract

The classroom serves as a vital setting for educational activities, functioning as the primary environment for knowledge transfer from teachers to students. In the digital age, the emergence of smart classrooms is revolutionizing traditional teaching methods and enhancing student learning through technology integration. During the COVID-19 pandemic, smart classrooms became essential for facilitating online education. This study aims to assess the effects of smart classrooms on learning outcomes, elucidate their advantages and disadvantages, and investigate the factors hindering their effective implementation. Conducted in the Rajanagaram Mandal of East Godavari district, Andhra Pradesh, this research involved a survey targeting both students and teachers across various high schools in the region. A total of 130 respondents (100 students and 30 teachers) were selected using simple random sampling methods. The study incorporated both qualitative and quantitative data, collected through primary sources, including observations, surveys, and semi-structured interviews, along with secondary data from government publications, academic journals, and internal records. The findings highlight the pros and cons of the smart classroom environment compared to traditional classrooms, providing valuable insights for future educational strategies.

Inference:

1. Impact Assessment: The study aims to evaluate how smart classrooms affect learning outcomes and the overall educational experience, which is crucial for understanding the effectiveness of technology in education.
2. Comprehensive Data Collection: By employing both qualitative and quantitative research methods, the study ensures a well-rounded perspective on the benefits and challenges associated with smart classrooms, enhancing the reliability of its conclusions.
3. Future Implementation: Identifying the shortcomings in the current implementation of smart classrooms provides critical insights that can guide policymakers and educators in optimizing the use of technology in education, ultimately improving the learning environment for students.

6. A Research on Machine Learning Methods and Its Applications

Abstract

Machine learning, a prominent subfield of artificial intelligence, emerged in the 1950s but saw limited research and development until the 1990s, when renewed interest led to significant advancements. As data continues to proliferate at an unprecedented rate, the need for effective analysis and processing has propelled the evolution of machine learning techniques. This science focuses on identifying the best models for new data based on insights gleaned from historical datasets. Consequently, the field of machine learning is expected to expand alongside the growing volume of data. This research paper delves into the history of machine learning, explores various methodologies employed in the field, and examines its diverse applications across industries. The primary aim of this study is to disseminate knowledge about machine learning, its methodologies, and its applications to researchers and practitioners, highlighting its increasing relevance in today's data-driven world.

Inference:

1. Historical Context and Evolution: The abstract provides a clear timeline of machine learning's development, illustrating its transition from limited research in the 1950s to

significant advancements in the 1990s and beyond, thereby contextualizing its importance in the present day.

2. Data-Driven Growth: The focus on the relationship between the proliferation of data and the advancement of machine learning emphasizes the necessity of this field in today's information-rich environment, suggesting that machine learning is crucial for effectively managing and interpreting large datasets.
3. Knowledge Dissemination: By aiming to share insights into machine learning's methodologies and applications, the study seeks to empower researchers and practitioners with the knowledge needed to leverage these technologies, promoting further innovation and application in various domains.

2.3 Inference Drawn

- **Enhanced Learning through Smart Classrooms**:
  Smart classroom models that integrate technologies like gesture recognition and AI-powered teaching tools significantly improve student learning outcomes and engagement compared to traditional methods.
- **Importance of Behavioral Data Analysis**:
  Analyzing teacher and student behavior in smart classrooms using models like SCTBA helps refine teaching strategies and classroom management, contributing to more efficient and personalized learning environments.
- **Challenges and Opportunities in Hand Gesture Recognition**:
  Machine learning-based gesture recognition enhances interactivity in smart education systems but still faces challenges such as low accuracy and high processing time, highlighting areas for future improvement.
- **Controlled Use of Mobile Technology**:
  Smart classroom applications that manage smartphone usage and automate attendance increase classroom efficiency and minimize distractions, ensuring technology is used constructively.
- **Smart Classrooms are Essential in the Digital Age**:
  The shift to smart classrooms during the pandemic underscores their importance and potential in delivering flexible, effective education, although implementation hurdles such as digital literacy and infrastructure must be addressed.

- **Machine Learning as a Backbone for Smart Education**:
  The growth of data in education demands robust machine learning applications to analyze and personalize learning, making ML a fundamental component of any smart classroom system like SmartLearn.

2.4 Comparison with Existing Systems

SmartLearn sets itself apart from existing platforms like Google Classroom, Moodle, and Microsoft Teams for Education by offering a more personalized and intelligent learning environment. While traditional platforms allow educators to upload study materials and create assignments, SmartLearn enhances this by organizing content dynamically based on the student's year, branch, semester, and subject. Its AI-based recommendation system suggests supplementary resources aligned with a student's progress, providing a customized and focused learning path that conventional systems do not typically support.

In terms of assessment, SmartLearn introduces a significant advancement through its AI-powered quiz generation module. Unlike Google Classroom or Moodle, where quizzes must be created manually or imported, SmartLearn automatically generates diverse question types using NLP, aligned with Bloom's Taxonomy. The system further adapts quiz difficulty based on a student's previous performance and offers real-time feedback, helping learners understand their strengths and areas for improvement. This automation reduces the workload for educators while making assessments more meaningful and interactive for students.

Live teaching experiences are also more interactive on SmartLearn compared to existing platforms. While others rely heavily on video conferencing tools for real-time lectures, SmartLearn incorporates machine learning-based hand gesture recognition, allowing teachers to write and draw on a virtual board during live classes. Though this feature is currently in early stages with lower accuracy, it adds an innovative, classroom-like experience to online education, something not typically offered in mainstream systems.

Chapter 3: Requirement Gathering for the Proposed System

3.1. Introduction to Requirement Gathering:

Requirement gathering is a crucial initial step in the software development lifecycle that focuses on identifying, analyzing, and documenting the needs and expectations of stakeholders for a new system. In the development of *SmartLearn: Intelligent Learning Platform*, this phase is especially important due to the dual-user nature of the platform—catering to both educators (admins) and students. The aim is to clearly understand what functionalities the system must support, such as uploading and accessing lectures, managing assignments and attendance, generating AI-based quizzes, enabling real-time live lectures using gesture control, and organizing educational content based on year, branch, semester, and subject.

This process involves various methods like interviews, surveys, brainstorming sessions, and studying existing platforms such as Google Classroom to identify limitations and opportunities for improvement. Gathering both functional requirements (what the system should do) and non-functional requirements (performance, security, scalability, etc.) helps in shaping a solution that is robust, scalable, and user-friendly. Ultimately, effective requirement gathering sets the foundation for successful system design and implementation by aligning the final product with user needs and technological advancements.

3.2. Functional Requirements:

User Authentication

- Users (teachers and students) must be able to register and log in securely.

Role-Based Access

- Teachers can upload lectures, assignments, notes, quizzes, attendance, etc.
- Students can only view/download/access content based on their year, semester, branch, and subject.

Content Upload & Management

- Teachers can upload URLs (e.g., YouTube videos), documents (PDF, PPT), and assignments.

Quiz Creation & Attempt

- Teachers can create quizzes with a unique ID.
- AI-generated quizzes using Google Gemini API based on random topics.
- Students can attempt quizzes and receive scores.

Gesture-Based Live Lectures

- Teachers can conduct live lectures using hand gesture recognition for writing/drawing.

Student Performance & Ranking

- System should store and display quiz scores and generate ranks based on performance.

Attendance Management

- Teachers can mark attendance and view records.
- Students can view their own attendance history.

Notification & News Posting

- Teachers can post important news/announcements visible to students.

Search and Filter Features

- Students can filter content by year, semester, branch, and subject.

3.3.Non-Functional Requirements:

Performance

- The system must respond quickly, especially for quiz loading, login, and content viewing.

Scalability

- The platform should handle a growing number of users and content without degrading performance.

Security

- Passwords must be encrypted.
- Only authorized users should access respective dashboards.
- Quiz cheating should be minimized (e.g., disabling copy-paste).

Usability

- User interface should be simple, clean, and intuitive for both students and teachers.

Accuracy

- AI-generated questions and gesture recognition must be reasonably accurate (to be improved in future work).

Reliability

- The system should be stable and recover gracefully from failures or crashes.

Maintainability

- Codebase should be modular and well-documented for future updates and debugging.

Compatibility

- The platform should work across devices (mobile, tablet, desktop) and browsers.

3.4.Hardware, Software, Technology and Tools Utilised:

A. Hardware Requirements:-

- **Processor:** Intel i5 or above
- **RAM:** Minimum 8 GB (Recommended 16 GB for smooth development and testing)
- **Storage:** Minimum 256 GB SSD
- **Graphics:** Integrated GPU for basic ML tasks (Dedicated GPU optional for better ML performance)
- **Camera:** Required for gesture recognition module
- **Internet Connection:** Stable broadband connection for real-time activities and YouTube integration

B. Software Requirements:-

- **Operating System:** Windows 10/11 or Ubuntu 20.04+
- **Code Editor:** Visual Studio Code / PyCharm
- **Database:** MySQL / Firebase Realtime Database
- **Web Server:** Apache / Node.js-based server
- **Browser:** Chrome, Firefox, Edge (latest versions)
- **Libraries/Packages:**
- NumPy, OpenCV, TensorFlow/Keras (for ML module)
- Flask / Django (for backend)
- React / Angular / HTML-CSS-JS (for frontend)

Techniques:-

- **Web Technologies:** HTML5, CSS3, JavaScript, Bootstrap, React.js
- **Backend Technologies:** Python (Flask or Django), Node.js (optional)
- **Database:** MySQL, Firebase (for user data and quiz scores
- **Cloud Services:** Google Drive / YouTube API for video storage and playback
- **AI/ML:** Google Gemini API for quiz generation, OpenCV + TensorFlow for gesture recognition
- **API Integration:** Google Gemini API for NLP-based question generation.

Tools Utilised:-

- **Version Control:** Git, GitHub
- **Project Management:** Trello / Jira / Google Sheets
- **Collaboration:** Google Meet / Zoom for team meetings
- **Design Tools:** Figma / Canva (for UI/UX designing)
- **Testing Tools:** Postman (for API testing), Selenium (for functional testing)
- **Documentation:** MS Word / Overleaf (for report writing), Google Docs

3.5. Constraints:

- **Gesture Recognition Accuracy:** Current machine learning model for hand gesture control has low accuracy, affecting usability in live lectures.
- **Video Dependency:** Lecture uploads rely on external YouTube URLs, limiting offline accessibility and content control.
- **Quiz Security:** Preventing cheating during quizzes is a challenge; no in-built restriction yet to stop copying or switching tabs.
- **API Limitation:** Dependency on Google Gemini API restricts quiz generation capabilities to the limitations of the external service.

Chapter 4: Proposed Design

This chapter outlines the comprehensive design of the **Smart Classroom system**, covering both conceptual and modular architecture. It begins with architectural diagrams that illustrate key system components—such as student and teacher interfaces, content management modules, and the air canvas feature—and their interactions. The chapter concludes with a Gantt chart that visualizes the project timeline, including task distribution, milestones, and progress tracking throughout the development cycle.

4.1. Block Diagram of the proposed system:

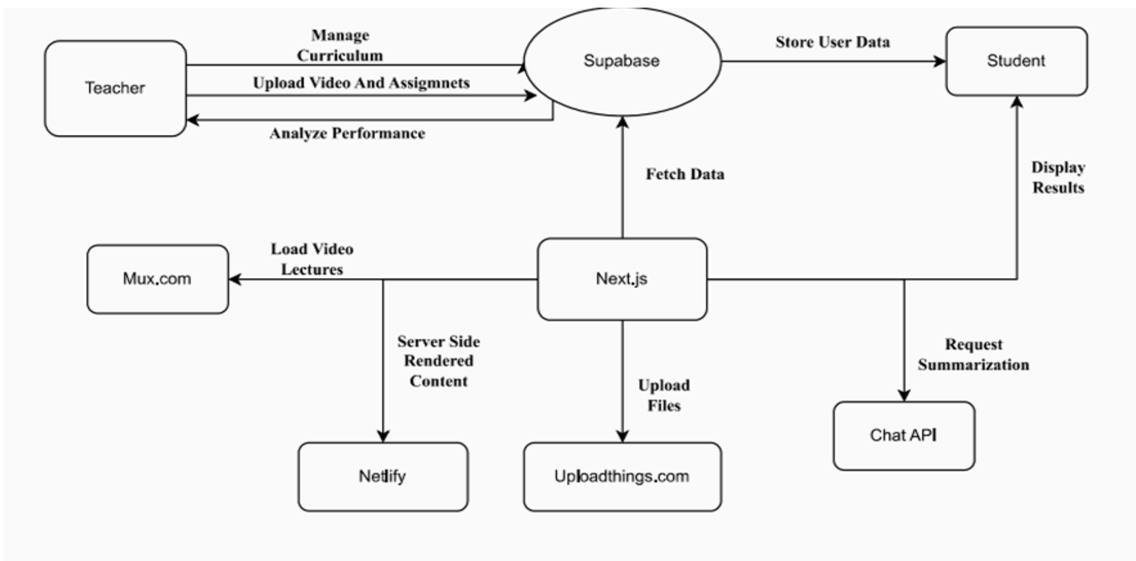

Fig. 4.1. Block Diagram of the System

1. Teacher
- Uploads videos and assignments to the system.
- Manages the curriculum.
- Analyzes student performance via data stored in the backend (Supabase).

2. Student
- Accesses content and views results.
- Triggers summarization requests for lecture content.

Core Backend Service

3. Supabase
- Acts as the database and backend-as-a-service.
- Stores user data (students, teachers, performance metrics, etc.).
- Fetches and manages all application-related data for rendering and analytics.

- Enables interaction between teachers and students through data storage.

Main Web Framework

4. Next.js

- Serves as the central application server.
- Handles server-side rendering of content.
- Communicates with:
- Supabase to fetch/store data.
- Mux.com to load and play video lectures.
- Uploadthings.com for file uploads.
- Netlify to deploy and serve frontend.
- Chat API to generate lecture summarizations.

Supporting Services

5. Mux.com

- Used for hosting and streaming video lectures uploaded by teachers.

6. Uploadthings.com

- A service that handles file uploads, such as lecture materials, images, documents, etc.

7. Netlify

- Hosts the Next.js application and delivers frontend via server-side rendered content.

8. Chat API

- Accepts student requests to summarize lecture content and delivers AI-generated summaries.

4.2. Modular diagram of the system:

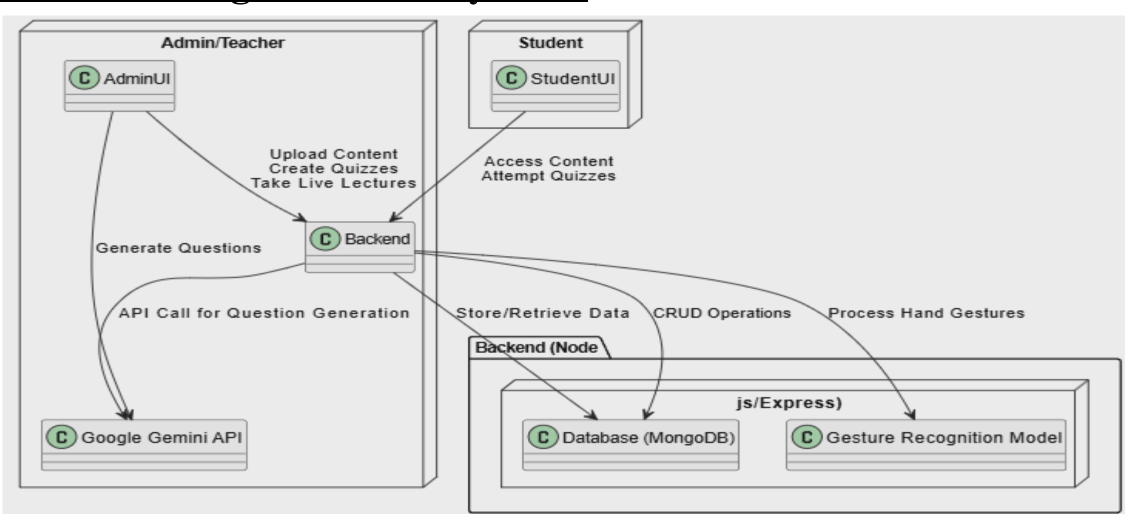

Fig.4.2 Modular Diagram

Admin/Teacher Side

- **AdminUI:** This is the front-end interface for teachers and admins.
- **Functions:**
    1. Upload content (like notes, videos, etc.)
    2. Create quizzes
    3. Take live lectures
- **Interaction with Backend:** AdminUI sends requests to the **Backend** to upload content, create quizzes, and manage live lectures.
- **Google Gemini API (for question generation):**
    - When an admin/teacher wants to generate quiz questions, the **Backend** sends an API call to **Google Gemini API**.
    - The Gemini API generates quiz questions automatically based on input.

2. Student Side

- **StudentUI:** This is the front-end interface for students.
- **Functions:**
    1. Access study content uploaded by teachers
    2. Attempt quizzes generated by teachers or AI
- **Interaction with Backend:** StudentUI communicates with the **Backend** to fetch content and quizzes.

3. Backend (Node.js/Express)

- **Main Control Center** of the system.
- **Functions:**
    1. Receive requests from AdminUI and StudentUI Make API calls to Google Gemini API.
    2. Store and retrieve data from **Database (MongoDB)**
    3. Process hand gestures using **Gesture Recognition Model**

- **CRUD Operations:**
  Create, Read, Update, Delete operations on study material, quizzes, student attempts, etc.

4. Database (MongoDB): Stores all the system data:

1. Uploaded content
2. Quizzes
3. Student attempts
4. User information (students and teachers)

5. Gesture Recognition Model

- A separate module that processes **hand gestures**.
- Example use case: In a live lecture, students might raise their hand virtually using hand gestures detected by the system.

4.3.Detailed Design (Flowchart)

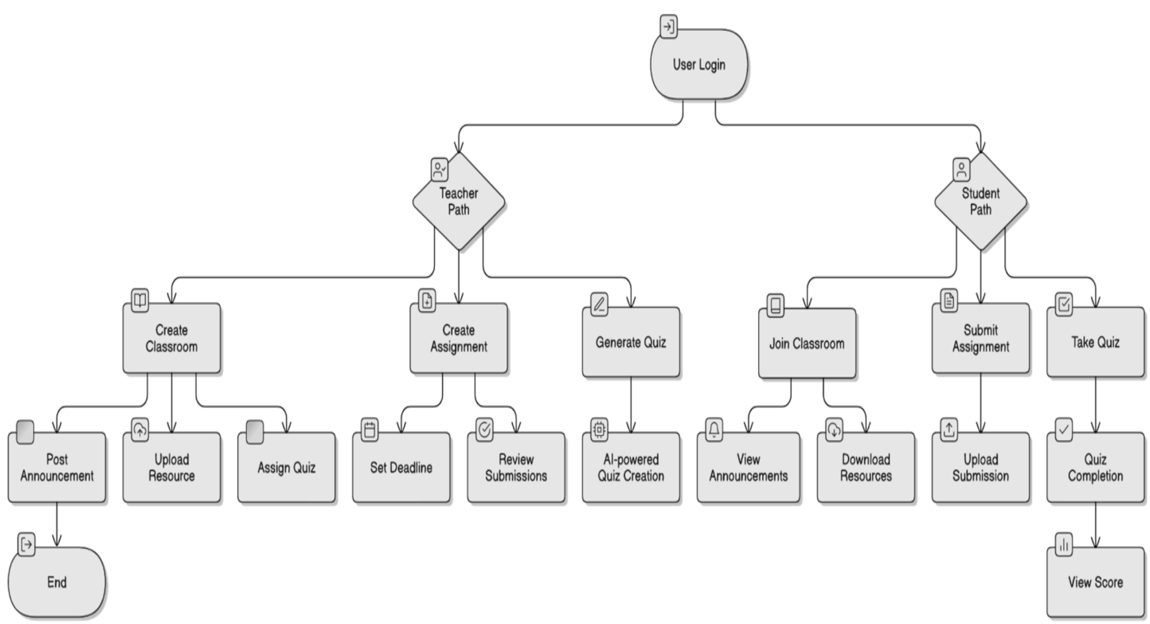

Fig 4.3 Detailed Design of the System

1. Start: User Login: Every user (whether a Teacher or a Student) must log in first.

2. Path Split: Teacher or Student

1. After login, user chooses a path:
2. Teacher Path
3. Student Path

Teacher Path

Teacher has 3 major activities:

1. Create Classroom: Teacher can:
   - Post Announcements (share important information)
   - Upload Resources (study materials, notes, videos)
   - Assign Quiz (create and assign quizzes to students)
2. Create Assignment: Teacher can:
   - Set Deadline for the assignment
   - Review Submissions once students upload their work
3. Generate Quiz: Teacher can:
   - Use AI-powered Quiz Creation to generate quizzes automatically based on topics/content

Student Path

Student has 3 major activities:

1. Join Classroom:
   - View Announcements (posted by teacher)
   - Download Resources (study materials provided)

2. Submit Assignment:
   - Upload Submission (upload completed assignments)

3. Take Quiz:
   - Attempt Quiz

○ After completing, they can View Score

End: Once actions are completed (posting, uploading, submitting, or taking quizzes), the flow ends.

4.4. Project Scheduling & Tracking using Time line / Gantt Chart:

The Gantt chart of our project where we worked for the whole semester to create this model is shown in a timeline pattern. It is the most important part to think and design the planning of your topic and so we planned our work like the gantt chart shown.

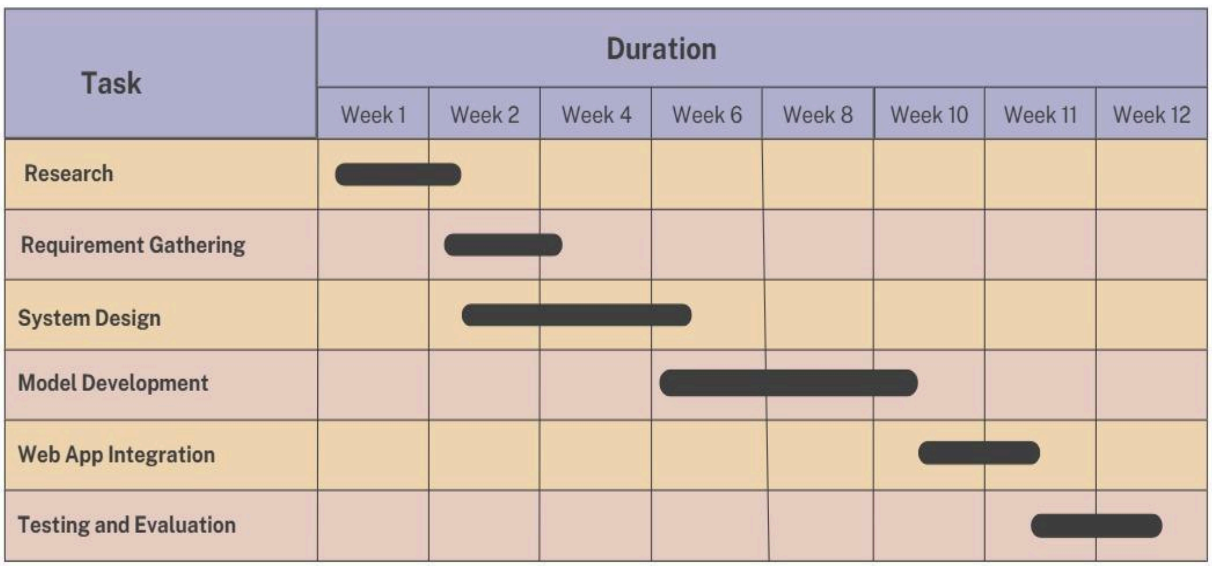

Fig 4.4 : Gantt chart

Chapter 5: Implementation of the Proposed System

5.1.Methodology employed for development:

1. Data Collection

- To implement gesture recognition for the SmartLearn platform, a dataset was required consisting of various hand gestures like writing, drawing, and basic shapes (e.g., circles, triangles). Data was collected using a webcam in real-time under different lighting conditions, angles, and hand orientations to ensure robustness.
- In the case of quiz generation, question data was fetched via integration with the Google Gemini API. The system takes a quiz topic input and generates relevant questions on the fly.
- For other platform modules (lectures, assignments, attendance, etc.), structured data is collected from the user input forms provided to the admin/teacher users.

2. Data Preprocessing

- Image Data (Gesture Recognition):
  - Background subtraction using OpenCV to isolate the hand.
  - Resizing and normalization of image frames.
  - Converting images to grayscale to reduce computational complexity.
  - Data augmentation (flipping, rotation) for better generalization.
- Text Data (Quiz Questions from Gemini):
  - Tokenization and filtering of API responses.
  - Removal of irrelevant or low-quality questions using rule-based filtering.
- Form Data:
  - Validation checks (e.g., required fields, proper formatting).
  - Automatic categorization based on year, semester, and branch.

3. Feature Extraction

- Gesture Recognition:
  - Contour-based features like convex hull, defect points.
  - HOG (Histogram of Oriented Gradients) features.
  - Keypoint tracking using MediaPipe and hand landmarks.

- Quiz Data:
  - Metadata like question type, complexity level, and keyword tagging.
  - Generated questions were vectorized using TF-IDF for categorization and relevance scoring.

4. Machine Learning Model Selection and Training

- Gesture Recognition:
  - Initially tested with Logistic Regression and Decision Trees.
  - Final model: Convolutional Neural Network (CNN) due to its better accuracy in image classification.
  - Trained using TensorFlow/Keras on labeled gesture data.

- Quiz Topic Understanding (Gemini Integration):
  - Google Gemini's language model API is used; no local model training required.
  - Queries are dynamically sent to generate topic-specific questions.

5. Model Evaluation and Performance Optimization

The models are evaluated using key metrics such as:

- Gesture Recognition Model:
  - Accuracy achieved: ~70% on the validation dataset.
  - Evaluation Metrics: Accuracy, Confusion Matrix, Precision & Recall.
- System Optimization:
  - UI optimized for fast loading using lazy loading techniques.
  - Database queries indexed for fast filtering by year, branch, etc.
  - Gemini quiz responses were cached to improve performance during quizzes.

5.2.Algorithms and Flowcharts for the respective modules developed:

1) **Admin side -**

a) Quiz Generation Module (using Google Gemini API):

Algorithm-

- Admin selects “Create Quiz”.

- Enter quiz topic/keywords.
- Send topic to Gemini API.
- Receive list of generated questions.
- Filter low-quality or irrelevant questions.
- Display questions to admin for review/edit.
- Save approved questions in the database with a unique Quiz ID.
- End.

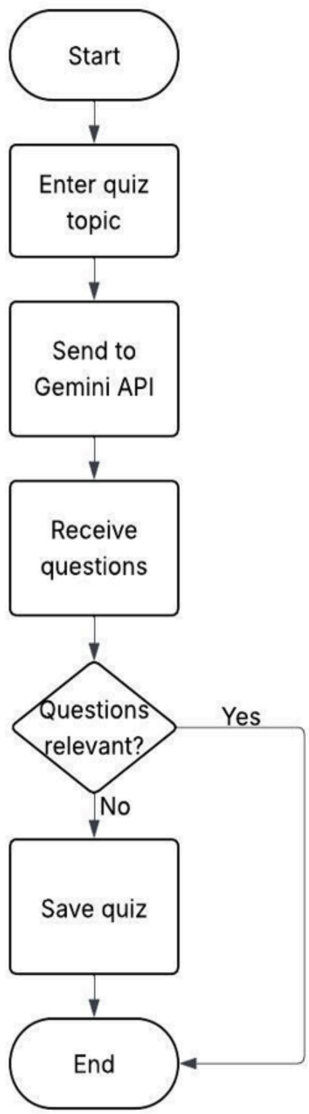

Fig 5.2.1: Flowchart of Quiz Generation Module (using Google Gemini API)

b) Lecture/Note Upload Module:

Algorithm -

- Admin selects “Upload Lecture”.
- Input metadata (title, subject, semester, branch, etc.).
- Paste YouTube video URL or upload file.
- Validate form data.
- Store the lecture info in the database.
- Notify students.
- End.

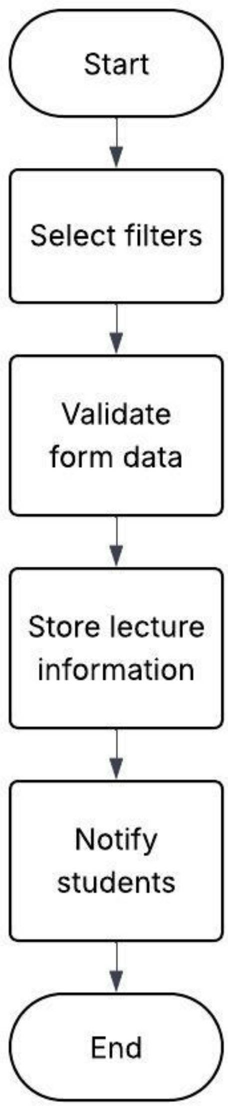

Fig 5.2.2: Flowchart of Lecture/Notes Upload Module

2)Student side-

Algorithm-

- **Students log into the system.**
- **Select filters (year, semester, branch, subject).**
- **Fetch and display matching lectures/notes/assignments.**
- **Click to view/download resources.**
- **Optionally take a quiz.**
- **View results and rank.**
- **End.**

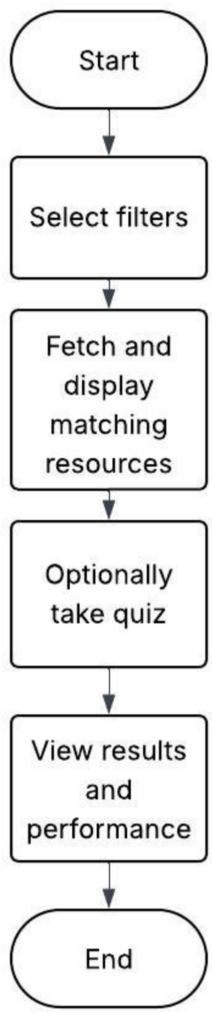

Fig 5.2.3: Flowchart of Lecture(Student Side)

5.3.Datasets source and utilisation:

In our project, we used two different datasets – one for Quiz Generation Using Gemini API and another for Lecture and Resource Metadata. Both datasets were publicly available and helped us train and test our machine learning and deep learning models.

A. Quiz Generation Using Gemini API:

- Source:
  - Dynamic input provided by the admin or teacher (topic/keywords).
  - Output dataset generated on-the-fly from the Google Gemini API which provides contextually relevant questions.
- Utilisation:
  - Each quiz question is stored in the database with a unique Quiz ID.
  - Teachers can edit or approve questions before assigning them to students.
  - Used to evaluate students' understanding of dynamically selected topics.

- Questions are filtered for duplicates and stored in the quiz table linked to a unique Quiz ID.
- Students access the quiz during assigned times; questions are auto-loaded and evaluated.

B. Lecture & Resource Upload Dataset:

- Source: Uploaded by teachers/admins through the platform:
  - YouTube URL**s** for lecture videos.
  - PDFs and Word Doc**s** for notes, assignments, and experiments.
- Utilisation:
  - Displayed to students based on filters they apply.
  - Helps organize and personalize the academic content shown on the dashboard.
  - Promotes subject-wise structured access to materials.
- Metadata Includes:
  - Year, Semester, Branch, Subject.
  - Title, Description, File URL, Upload Date.

Chapter 6: Testing of the Proposed System

6.1.Introduction to Testing :

Testing is a critical phase in the software development life cycle, especially for educational platforms like Smart Classrooms, where functionality, usability, and system stability are essential. In this project, the Smart Classroom system was designed to support interactive learning through features such as lecture video access, assignment submission, performance tracking, and the unique "air canvas" for gesture-based annotation. As such, thorough testing was essential to ensure that all components perform reliably under different usage scenarios. The testing process was structured to evaluate the system’s performance, accuracy of user interactions (e.g., gesture recognition), content delivery efficiency, and overall user experience for both students and teachers. It began with unit testing to validate individual modules like login authentication, content upload, and air canvas responsiveness, followed by integration and system testing to assess the platform's behavior as a whole. Special emphasis was placed on the responsiveness of real-time features and the accuracy of data displayed, such as attendance and assessment results. The goal was to deliver a stable, user-friendly educational tool that educators and learners can trust for consistent, effective teaching and learning experiences.

6.2.Types of tests Considered:

A. Pre testing phase

Before the final deployment of the Smart Classroom system, a pre-testing phase was carried out to assess the platform’s usability, performance, and functionality under real-world conditions. This phase involved a group of teachers and students who interacted with key features such as login access, content navigation, lecture video streaming, assignment uploads, and the "air canvas" gesture-based annotation tool. The purpose was to ensure that inputs—such as hand gestures, content filters by year/branch/subject, and assessment data—were accurately processed and reflected in the system. Feedback from participants highlighted areas for improvement, including gesture recognition sensitivity and UI responsiveness, leading to adjustments in model calibration and user interface design. Additionally, user insights were used to refine content categorization and improve the clarity of system navigation, ensuring a smoother and more intuitive experience for both teachers and students.

B. Beta-Testing Phase

Following the successful pre-testing phase, beta testing was conducted by our student team simulating both student and teacher roles to evaluate the Smart Classroom system in a real-time educational environment. Each team member performed key tasks such as uploading and accessing lecture videos, submitting and reviewing assignments, checking attendance records, and using the "air canvas" feature for live annotation. Since no external teachers were involved, we thoroughly tested the admin (teacher) functionalities ourselves to ensure usability, responsiveness, and feature accuracy. Our project mentor provided guidance and observed the process, offering valuable suggestions for improving gesture recognition and interface clarity. Based on our internal testing and feedback, several refinements were made to enhance the gesture-based drawing tool, optimize navigation, and improve the layout for better content management. The beta testing phase validated the system’s functionality, interactivity, and user-friendliness, confirming its readiness for deployment in an educational setting.

6.3.Various test case scenarios considered:

User Login and Registration

- Verified that new users (both student and teacher roles tested by our team) could successfully register and log in.
- Ensured appropriate error messages appeared when incorrect login credentials were used.

Content Upload and Access (Smart Classroom Functionalities)

- Tested the teacher (admin) side by uploading lecture videos, assignments, and attendance reports—ensuring uploads were smooth and categorized properly by academic year, branch, semester, and subject.
- On the student side, verified that uploaded materials were accessible, organized correctly, and downloadable.

"Air Canvas" Gesture Recognition (Hand-Pen Tool)

- Evaluated the real-time drawing feature using hand gestures.

- Tested different lighting conditions and hand movements to ensure the model could accurately recognize gestures.
- Checked for smooth drawing flow and ensured that annotations were visible and responsive.

System Performance

- Simulated multiple users logging in and accessing/uploading content simultaneously to assess system responsiveness and server stability.
- Tested load handling by uploading large files and switching between roles quickly.

Data Security and Access Control

- Ensured that login credentials and uploaded content remained secure.
- Verified that students could not access admin features and that each user's data was restricted appropriately

Chapter 7: Results and Discussions

7.1.Screenshot of Use Interface(UI) for the system:

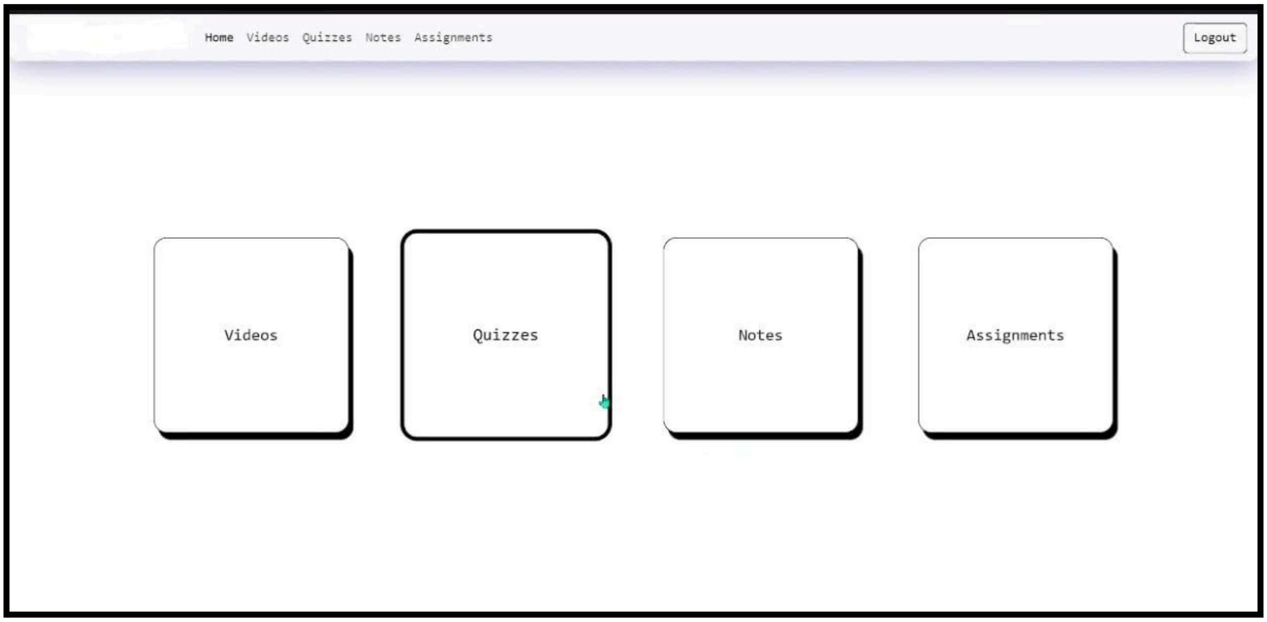

Fig 7.1.1 Home page

The homepage serves as the entry point to the SmartLearn platform, providing an overview of its core functionalities.

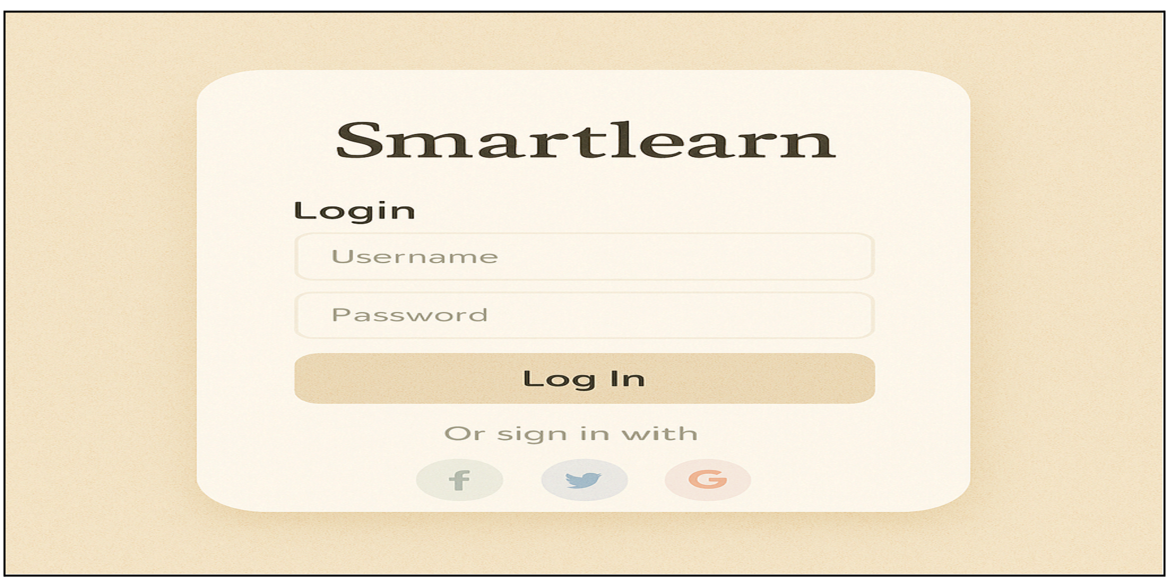

Fig 7.1.2: Login page

The login page provides a secure gateway to the platform, requiring user credentials for access.

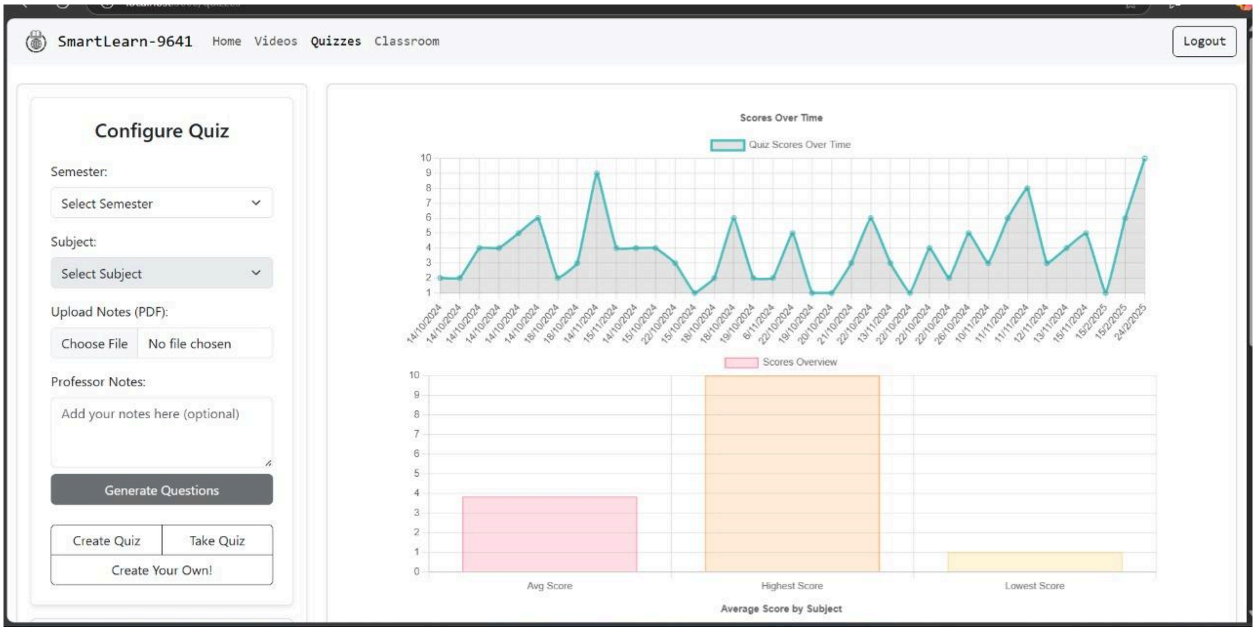

Fig 7.1.3: Quiz Management Dashboard

The dashboard is designed with an intuitive layout to enhance usability and encourage interactive learning and continuous assessment.

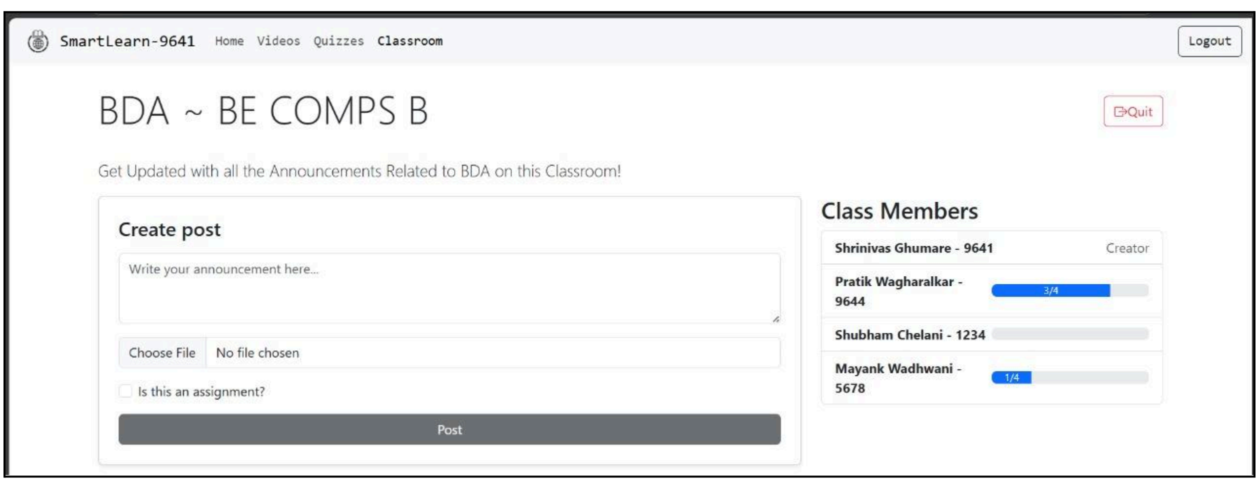

Fig 7.1.4: Classroom Announcement Interface

This simple and clean design ensures effective communication between class members and helps manage coursework efficiently.

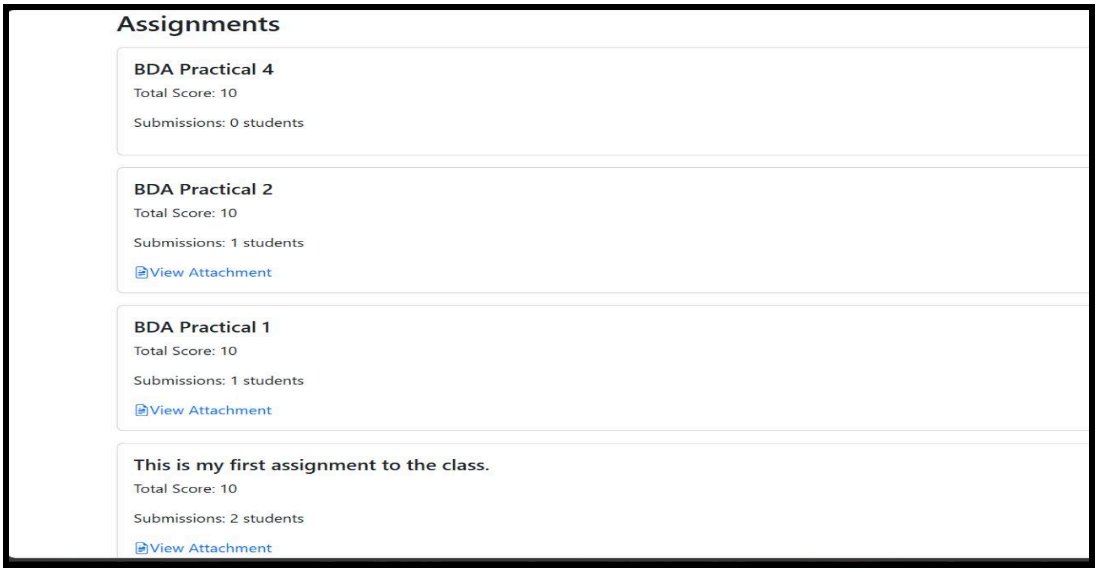

Figure 7.1.5: Assignments Section

The interface provides a clean and organized way for students to access assignments, check submission status, and for faculty or admin to manage and track the assignment progress easily.

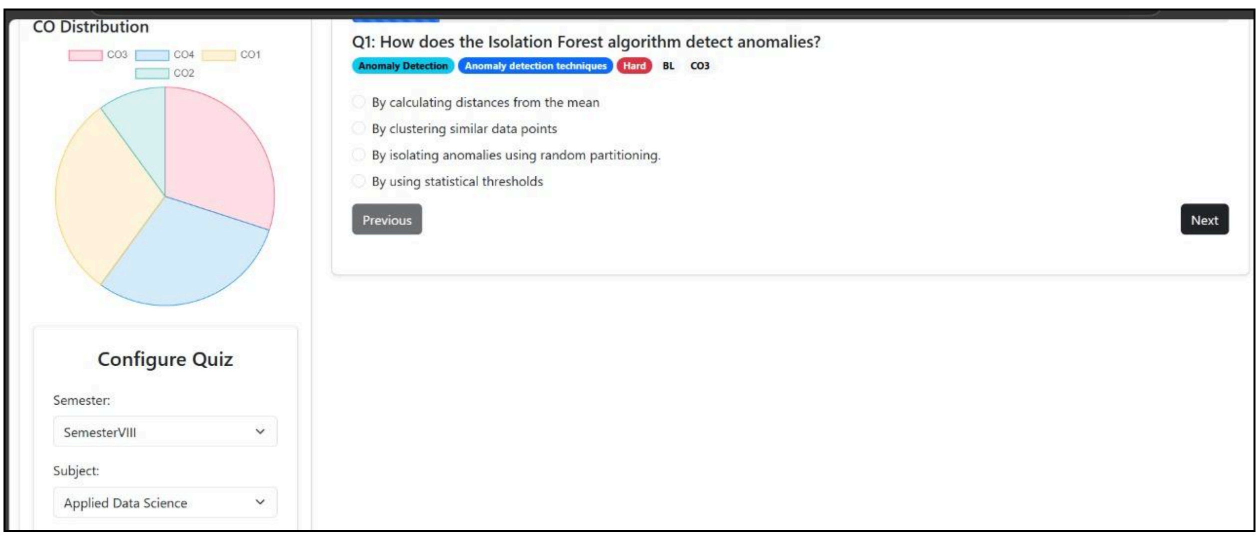

Figure 7.1.6 : Quiz Interface

This screen shows the quiz interface for the "Applied Data Science" subject. A multiple-choice question about the Isolation Forest algorithm is displayed, with "Previous" and "Next" navigation buttons. On the left, there's a pie chart showing the Course Outcome (CO) distribution and options to configure the quiz by semester and subject.

7.2. Performance Evaluation Measures:

1. Precision: Precision is one indicator of a machine learning model's performance – the quality of a positive prediction made by the model. Precision refers to the number of true positives divided by the total number of positive predictions (i.e., the number of true positives plus the number of false positives). The formula is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

where:

TP = True Positives,

FP = false Positives.

Model's Precision score = 75%

2. Recall: The recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect positive samples. The higher the recall, the more positive samples detected. The formula is:

$$\text{Recall} = \frac{TP}{TP + FN}$$

where:

TP = True Positives,

FN = false Negatives.

Critical for lung cancer detection, where missing a true case (false negative) can delay treatment.

Model's Recall Score = 78%

3. F1-Score: The F-score (also known as the F1 score or F-measure) is a metric used to evaluate the performance of a Machine Learning model. It combines precision and recall into a single score. The formula is:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Balances the trade-off between precision and recall, offering a single score to evaluate performance.

Model's F1- Score = 76%

4. Accuracy: The ratio of correctly predicted observations to the total observations.The Formula is :

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Model's accuracy= 79%

7.3. Input Parameters/Features considered:

Input Parameters/Features Considered for Air Canvas ML Model:

- **Image/Video Input**:
  The model processes frames captured from the webcam in real-time, detecting the hand gestures or movements in the air.
- **Preprocessing**:
  Each frame is resized, normalized, and converted to a color space (such as HSV) to better segment the hand from the background.
- **Hand Detection and Tracking**:
  Features like **contour detection**, **convex hull**, and **centroid tracking** are used to accurately follow hand motion.
  In some models, fingertip detection or color segmentation techniques (using gloves or markers) are applied for more precision.
- **Gesture Recognition**:
  Key parameters such as **finger position**, **motion trajectory**, and **gesture stability** are extracted to recognize drawing actions like freehand curves, clicks, and line segments.
- **Feature Importance**:
  Movement smoothness, hand stability, and pixel intensity changes are crucial for distinguishing between intended drawing gestures and random movements.

- **Robust to Noise**:
  The system applies morphological operations and noise filtering to reduce false detections caused by lighting variations or background distractions, ensuring better accuracy between **70–80%**.

7.4. Inference Drawn:

The developed Air Canvas ML system offers an innovative solution compared to traditional input methods like physical drawing tablets or touchscreens. Unlike conventional systems, which require direct contact or expensive hardware, our model leverages real-time hand-tracking and gesture recognition to allow drawing in the air, making it more accessible and cost-effective. Using computer vision techniques, the system achieves an accuracy of around **70–80%**, enabling smooth and efficient virtual interaction. Overall, the Air Canvas project provides a scalable, affordable, and intuitive alternative for creative applications, online teaching, and virtual writing interfaces.

Chapter 8: Conclusion

8.1.Limitations:

- While the Smart Classroom system introduces several innovative features, it has a few notable limitations. The effectiveness of the **air canvas (hand-pen)** depends on consistent lighting and background simplicity, which can affect gesture accuracy and user experience.
- **Content quality** is another challenge, as the system relies on manual uploads by student users during development; this may lead to inconsistencies in formatting, clarity, or completeness of educational materials.
- The system also **lacks automated content verification**, meaning errors in subject categorization or file naming can hinder easy access to resources. While designed for scalability, **performance under high traffic or institutional use** has not been fully tested. Moreover, **user experience issues** such as layout design, navigation consistency, and responsiveness on various devices still require refinement.

8.2.Conclusion:

In an age where digital transformation is reshaping education, the Smart Classroom system developed in this project demonstrates how technology can enhance teaching and learning experiences, especially in engineering education. By integrating features like organized content access, student-specific dashboards, and a unique air canvas tool powered by machine learning, the system promotes interactive, flexible, and remote-friendly learning. Despite being developed and tested entirely by students, the platform provides core functionalities such as lecture video uploads, assignment tracking, and live teaching support. Its web-based structure ensures accessibility, while the hand-pen feature mimics real-time blackboard interaction. Although some limitations exist in content quality and feature refinement, this project showcases a scalable, innovative approach to modern education, highlighting the potential of AI and smart interfaces in bridging gaps in traditional classroom environments

8.3.Future Scope:

A. Improving ML Accuracy for Gesture Recognition

- Currently, the hand gesture recognition model used in live lectures has limited accuracy. Future work will focus on enhancing the model's performance by experimenting with advanced algorithms such as CNNs, RNNs, and Transformer models. Incorporating more extensive and diverse training datasets will further improve the system's accuracy and reliability.

B. Enhancing UI/UX Design

- To improve the user experience, SmartLearn's interface will be redesigned with a more intuitive, user-friendly layout. Focus will be placed on streamlining navigation, improving content organization, and ensuring accessibility across various devices. User feedback will be continuously incorporated to enhance the platform's functionality and visual appeal.

C. Preventing Cheating During Quizzes

- To strengthen the platform's anti-cheating mechanisms, future updates will implement AI-powered proctoring with real-time monitoring, face recognition, and browser activity tracking. Additionally, enhanced copy-prevention techniques and randomized question sequences will be applied to minimize the chances of cheating during online assessments.

D. AI-Powered Personalized Learning Paths

- SmartLearn will incorporate AI-based adaptive learning models to offer personalized content recommendations and dynamic learning paths. By analyzing student performance and engagement, the system can provide customized learning experiences, ensuring that students receive content aligned with their skill level and learning pace.

E. Mobile Application Development

- To extend accessibility and improve engagement, a mobile application version of SmartLearn will be developed. This will ensure that students and teachers can access course materials, attend live lectures, and participate in quizzes anytime, anywhere.

F. Multilingual Support and Localization

- To cater to a diverse student base, future versions of SmartLearn will include multilingual support and content localization. This will ensure that students from different regions can access content in their preferred language, enhancing inclusivity and learning outcomes.

References

1. A. M. Al-Sharideh, T. H. Al-Rabadi, and M. S. Al-Hassan, "The impact of smart classrooms on the performance of students in higher education," *IEEE Access*, vol. 8, pp. 88833-88842, 2020. doi: 10.1109/ACCESS.2020.2992502.
2. A. H. S. S. J. O. Chik, "An implementation of a smart classroom system using IoT technology," *2018 IEEE 9th International Conference on Awareness Science and Technology (iCAST)*, pp. 249-254, 2018. doi: 10.1109/ICAwST.2018.8591401.
3. J. Wu, H. Yang, J. Yang, and Y. Zhang, "Development of smart classroom system based on Internet of Things," *2019 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pp. 58-63, 2019. doi: 10.1109/SmartIoT.2019.00017.
4. T. Zhang, L. Wang, and Y. Guo, "The application of artificial intelligence in smart classrooms," *2019 IEEE International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 123-126, 2019. doi: 10.1109/ICAIBD.2019.8833020.
5. Y. Hu, Y. Wang, H. Wang, and Z. Jiang, "Research on the application of big data technology in smart classrooms," *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1-6, 2019. doi: 10.1109/BigComp.2019.8641985.
6. H. Zhang, "Exploration on the construction of smart classrooms based on 5G technology," *2020 IEEE International Conference on Computer Engineering and Technology (ICCET)*, pp. 176-180, 2020. doi: 10.1109/ICCET49364.2020.00048.
7. M. M. K. Alali, M. H. Ibrahim, and A. A. A. Al-Hussein, "A smart classroom management system using machine learning," *2020 IEEE International Conference on Cloud Computing and Artificial Intelligence (CCAI)*, pp. 162-167, 2020. doi: 10.1109/CCAI51114.2020.00038.
8. K. Lee and K. Kim, "Design and implementation of a smart classroom management system using machine learning," *2020 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 109-113, 2020. doi: 10.1109/ICCCS49541.2020.9053688.
9. P. K. Gupta, S. Y. Lee, and A. J. K. Y. Liu, "Enhancing learning through smart classrooms: A case study in engineering education," *2020 IEEE 22nd International Conference on Advanced Learning Technologies (ICALT)*, pp. 74-78, 2020. doi: 10.1109/ICALT49669.2020.00021.
10. S. M. Wibawa, D. A. Handayani, and H. N. Aini, "Development of an e-learning platform using a collaborative learning approach for higher education," *Journal of Physics: Conference Series*, vol. 1193, no. 1, pp. 1–7, 2019.
11. R. Sharma and P. Sharma, "AI-based personalized learning: Enhancing education through adaptive learning technologies," *International Journal of Artificial Intelligence and Education*, vol. 36, no. 2, pp. 123–135, 2022.
12. T. Zhang and A. Shah, "Gesture recognition using convolutional neural networks for interactive teaching," *IEEE Access*, vol. 9, pp. 23456–23464, 2021.
13. M. A. Jadhav and S. S. Lathkar, "Automated question generation using NLP and machine learning techniques," *International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 456–461, 2020.

14. Y. Xu, L. Wang, and H. Liu, "Improving online learning experiences through adaptive assessment and intelligent feedback," *Journal of Educational Technology Systems*, vol. 49, no. 4, pp. 497–513, 2021.
15. G. V. Rao and K. B. Reddy, "Cloud-based learning management systems: A comprehensive review," *International Journal of Cloud Computing and Education (IJCCE)*, vol. 15, no. 3, pp. 245–260, 2021.
16. H. Alqahtani and A. Rajkhan, "E-learning critical success factors during the COVID-19 pandemic: A comprehensive analysis," *Education and Information Technologies*, vol. 26, pp. 7053–7075, 2021.
17. P. Mukherjee and R. Kumar, "AI-powered proctoring: Enhancing security in online assessments," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 4, pp. 129–140, 2022.
18. A. K. Yadav and S. Tiwari, "Virtual labs and simulations for enhancing engineering education: A case study," *IEEE Global Engineering Education Conference (EDUCON)*, pp. 312–319, 2021.

Appendix

1. Paper I details :-

a. Paper 1 :

SmartLearn: Intelligent Learning Platform

Mayank Wadhwani
dept.of computer engineering
vivekanand education society institute of technology
Mumbai,India
2021.mayank.wadhwani@ves.ac.in

Shrinivas Ghumare
dept.of computer engineering
vivekanand education society institute of technology
Mumbai, India
2021.shrinivas.ghumare@ves.ac.in

Pratik Wagharalkar
dept.of computer engineering
vivekanand education society institute of technology
Mumbai, India
2021.pratik.wagharalkar@ves.ac.in

Shubham Chelani
dept.of computer engineering
vivekanand education society institute of technology
Mumbai, India
2021.shubham.chelani@ves.ac.in

Mrs. Prerna Solanke
Prof. *dept.of computer engineering*
vivekanand education society institute of technology
Mumbai, India
prerna.solanke@ves.ac.in

Abstract—SmartLearn: Intelligent Learning Platform is an advanced web-based system designed to improve the teaching and learning process by integrating artificial intelligence and machine learning technologies. The platform provides a dynamic environment where teachers can upload lectures, assignments, notes, and experiments, take live lectures, and conduct quizzes with unique IDs and AI-generated questions using the Google Gemini API. Students can access these resources based on their year, semester, branch, and subject. A notable feature is the use of hand gesture recognition during live lectures, enabling teachers to write and draw shapes, although the current accuracy of the machine learning algorithms used is low, with plans for enhancement. Additionally, the platform aims to introduce anti-cheating measures by removing the copy option during quizzes. Future developments will focus on improving algorithm accuracy, enhancing the platform's UI/UX design, and ensuring a secure, scalable, and efficient learning environment.

Keywords—Smart Classroom, Engineering Education, Digital Learning, Interactive Learning

I. Introduction

In the era of digital transformation, the education sector has witnessed a significant shift toward online learning platforms that offer flexible and interactive learning experiences. **SmartLearn: Intelligent Learning Platform** is a web-based system designed to address the growing need for a comprehensive and efficient learning management system (LMS) that caters to the requirements of both teachers and students. Inspired by platforms like Google Classroom, SmartLearn provides an all-in-one solution where teachers can upload and manage course content such as lecture videos, assignments, experiments, notes, and attendance records. Students, in turn, can seamlessly access these resources based on their year, semester, branch, and subject, ensuring a personalized and organized learning experience.

A key feature that sets SmartLearn apart is the integration of **artificial intelligence (AI)** and **machine learning (ML)** to enhance the teaching and assessment process. Teachers can conduct live lectures where they can write or draw shapes using hand gestures, facilitated by ML algorithms. Although the current accuracy of the gesture recognition system is low, continuous improvements are planned to enhance its reliability. Additionally, SmartLearn leverages the **Google Gemini API** to generate random quiz questions on various topics, allowing teachers to create quizzes effortlessly. Each quiz is assigned a unique ID to prevent duplication and ensure a streamlined assessment process.

To maintain academic integrity, SmartLearn plans to introduce anti-cheating mechanisms, such as removing the copy option during quizzes. The platform's future development roadmap includes improving the accuracy of ML algorithms, enhancing the user interface and user experience (UI/UX) to ensure a more intuitive design, and strengthening security measures to provide a scalable and secure learning environment. By integrating innovative technologies and focusing on continuous improvements, SmartLearn aims to redefine the online education landscape and foster a more engaging and efficient learning experience for all users.

II. Literature Review

The development of online learning platforms has gained significant momentum in recent years, especially with the rise of remote education and the increasing demand for flexible learning environments. Several platforms have emerged that offer comprehensive solutions for managing course content, conducting assessments, and facilitating communication between teachers and students. Among these, **Google Classroom** has established itself as one of the most popular platforms, providing features such as content uploading, assignment management, and quiz creation. However, Google Classroom lacks AI-powered quiz generation and live gesture recognition, which SmartLearn aims to address.

Moodle, another widely used learning management system (LMS), offers a highly customizable open-source platform that allows educators to create and manage courses. While Moodle supports plugin integration and extensive customization, it does not natively support features such as AI-based question generation or real-time gesture recognition during live lectures. **Edmodo** is another platform that focuses on creating an interactive learning environment by enabling collaboration and content sharing. However, Edmodo's functionality is primarily centered around content delivery and lacks the AI-powered features that SmartLearn introduces.

Recent advancements in **AI and machine learning** have led to the development of automated question generation systems, enhancing the assessment process. Studies have shown that models such as **Google Gemini** can generate contextually relevant questions for various topics, reducing the workload on educators and improving the quality of assessments. Furthermore, gesture recognition systems using convolutional neural networks (CNNs) and deep learning algorithms have shown promise in enabling intuitive interaction during live sessions, although current accuracy levels remain a challenge.

In terms of anti-cheating mechanisms, platforms such as **ProctorU** and **ExamSoft** offer remote proctoring solutions that detect suspicious activities during online assessments. These platforms employ AI-powered monitoring and restrict certain actions, such as copying and switching between windows. SmartLearn aims to incorporate similar anti-cheating measures by disabling the copy option during quizzes to maintain academic integrity.

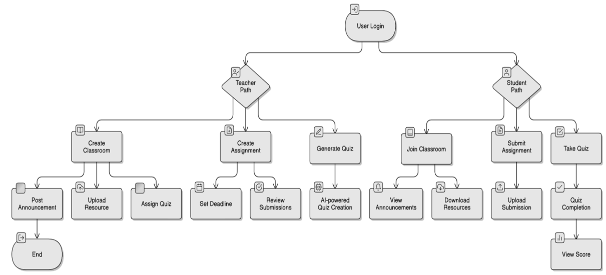

Fig. 2.1 - User Interaction Flowchart

While existing platforms provide essential features for online education, SmartLearn distinguishes itself by integrating AI-based quiz generation, real-time gesture recognition, and advanced anti-cheating mechanisms, making it a more intelligent, secure, and efficient learning platform. The combination of these technologies positions SmartLearn as a comprehensive solution that addresses the limitations of traditional learning management systems.

III. Methodology

The development of **SmartLearn: Intelligent Learning Platform** follows a systematic approach that integrates web technologies, artificial intelligence, and machine learning to create a dynamic and interactive learning environment. The methodology involves multiple stages, including system design, module development, and implementation of key features to ensure a scalable and secure platform.

A. System Design and Architecture

The platform is developed using a **three-tier architecture** comprising:

- **Frontend:** Developed using HTML, CSS, JavaScript, and React.js to ensure a responsive and intuitive user interface.
- **Backend:** Powered by Node.js and Express.js to handle API requests, manage user data, and ensure seamless interaction between the frontend and database.
- **Database:** MongoDB is used as the database to store and manage user profiles, course content, assignments, quiz results, and attendance records.

B. User Roles and Access Control

- **Admin (Teacher):** Teachers can upload lectures, assignments, notes, experiments, and attendance. They can also conduct live sessions, create quizzes, and generate random quiz questions using the Google Gemini API.

- **Student:** Students can access uploaded content, participate in live sessions, and take quizzes based on their year, semester, branch, and subject.

C. Quiz Generation Using Google Gemini API

To automate the process of quiz creation, the platform integrates the **Google Gemini API** to generate random questions on various topics. Teachers can specify the subject and difficulty level, and the API returns a set of contextually relevant questions. Each quiz is assigned a **unique quiz ID** to prevent duplication and ensure efficient management of assessments.

D. Live Lecture and Gesture Recognition

During live lectures, teachers can write or draw shapes using **hand gestures** that are recognized through a machine learning-based gesture recognition system. The system uses **convolutional neural networks (CNNs)** and **OpenCV** to detect and interpret hand movements, allowing for intuitive interaction. Although the current accuracy of gesture recognition is low, future iterations will aim to enhance performance through model optimization and data augmentation.

E. Anti-Cheating Measures in Quizzes

To maintain academic integrity during quizzes, SmartLearn incorporates anti-cheating mechanisms by disabling the copy-paste option and restricting external browser navigation. Additional monitoring features will be introduced in future updates to further reduce instances of cheating.

F. UI/UX Design and Enhancement

The platform's user interface is designed with a focus on simplicity and ease of navigation. A modular approach is followed to ensure that both teachers and students can access relevant features without unnecessary complexity. Future plans include enhancing the UI/UX design to improve accessibility and ensure a more engaging user experience.

G. Performance Evaluation and Testing

The platform undergoes rigorous testing, including:

- **Unit Testing:** To validate individual modules and ensure functionality.
- **Integration Testing:** To verify seamless communication between frontend, backend, and database.
- **User Acceptance Testing (UAT):** To gather feedback from potential users and refine the system based on their suggestions.

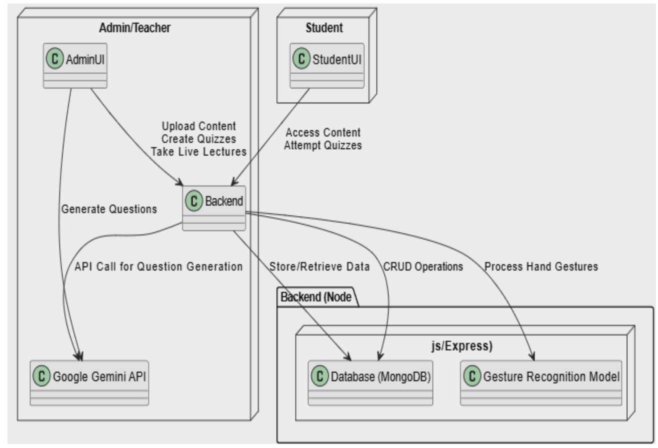

Fig. 3.1 - Block Diagram

The Block Diagram depicts the simplified flow of data and interactions between system components. It demonstrates how Admins and Students interact with the backend to perform various actions such as content upload, quiz generation, and accessing materials. The backend communicates with the database, Google Gemini API for question generation, and the ML model for gesture recognition, ensuring seamless integration of all system features.

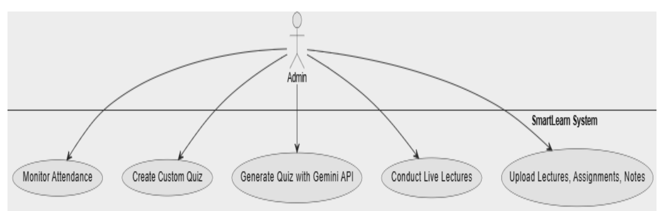

Fig. 3.2 - Admin Use Case Diagram

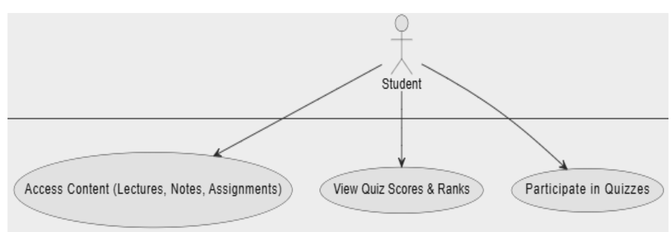

Fig. 3.3 - Student Use Case Diagram

The Use Case Diagram illustrates the different functionalities available to the two types of users: Admin (Teachers) and Students. Admins can upload content, generate quizzes, and conduct live sessions, while students can access course content, attempt quizzes, and view scores.

This diagram highlights the key system functionalities and how users interact with them.

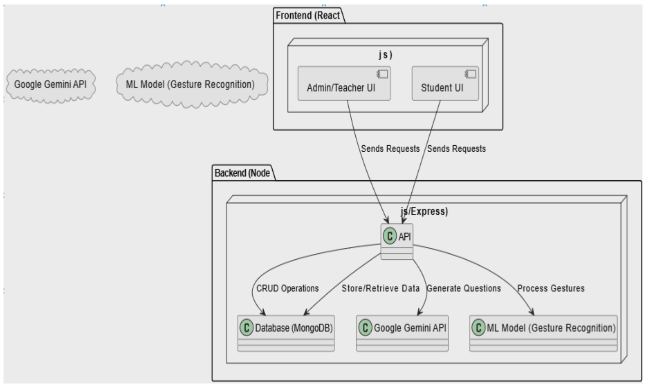

Fig. 3.4 - System Architecture Diagram

The System Architecture Diagram provides an overview of the platform's structure, showcasing the interaction between the frontend (React.js), backend (Node.js/Express), and database (MongoDB). It highlights how teachers and students interact with the platform through respective UIs, while the backend handles API requests, communicates with the Google Gemini API for quiz generation, processes gestures through the ML model, and manages data storage and retrieval.

IV. Features and Implementation

Features

A. Admin/Teacher Features

- **Content Management:** Upload lectures, assignments, experiments, notes, and attendance.
- **Live Lectures with Gesture Recognition:** Conduct live lectures with hand gesture recognition for writing and drawing shapes.
- **Quiz Generation with Google Gemini API:** Automatically generate quizzes on random topics by integrating the Google Gemini API.
- **Custom Quiz Creation:** Manually create personalized quizzes with a unique quiz ID for different topics.
- **Performance Analytics:** Track student performance, quiz scores, and ranks to analyze individual and class-level progress.

B. Student Features

- **Access to Course Material:** View and download uploaded lectures, notes, assignments, and other resources.
- **Participate in Quizzes:** Attempt quizzes created by teachers and view quiz scores and ranks.
- **Notifications & News:** Receive updates on assignments, quiz schedules, and course-related information.
- **Video Lecture Access:** Access YouTube URLs uploaded by teachers for recorded video lectures.

C. Security & Anti-Cheating Features

- **Disabling Copy-Paste During Quizzes:** Prevents copying and pasting answers during quizzes to reduce cheating.
- **Secure Authentication:** Ensures secure login and role-based access control to protect data.

Implementation

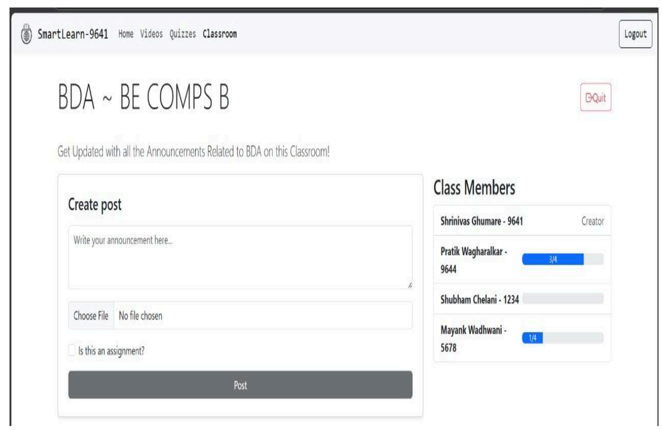

Fig. 4.1 - Classroom Management

SmartLearn provides a structured classroom environment where educators can create courses, post assignments, and manage student interactions efficiently. Teachers can track student progress, provide personalized feedback, and manage class discussions in real time. The platform includes attendance tracking, participation analysis, and automated grading for quizzes and assignments, reducing administrative workload. Collaborative learning is supported through discussion forums and group project management tools that enable task assignment, progress tracking, and real-time collaboration. The analytics dashboard offers insights into student performance trends, helping teachers identify learning gaps and personalize instruction. By centralizing these features, SmartLearn enhances the digital classroom experience, promoting academic success and student engagement.

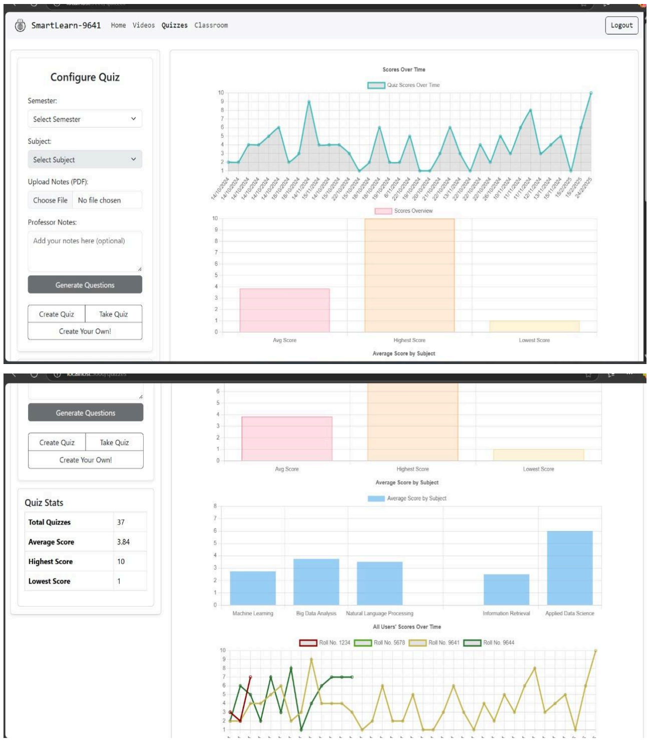

Fig. 4.2 - AI-Powered Quiz Generation

The AI-driven quiz generation module in **SmartLearn** uses NLP to create assessments aligned with Bloom's Taxonomy, ensuring a thorough evaluation of cognitive skills. Teachers can input topics or course materials, and the system automatically generates various question types, including multiple-choice, short-answer, and essay-based questions. SmartLearn analyzes student responses in real time, offering immediate feedback and identifying areas for improvement. The adaptive assessment feature personalizes quizzes based on previous student performance, providing tailored challenges that enhance engagement and retention. AI-powered analytics offer insights into learning patterns, enabling educators to refine instructional strategies. By categorizing questions based on difficulty and cognitive skills, SmartLearn ensures balanced and effective assessments, optimizing student learning experiences while reducing educator workload.

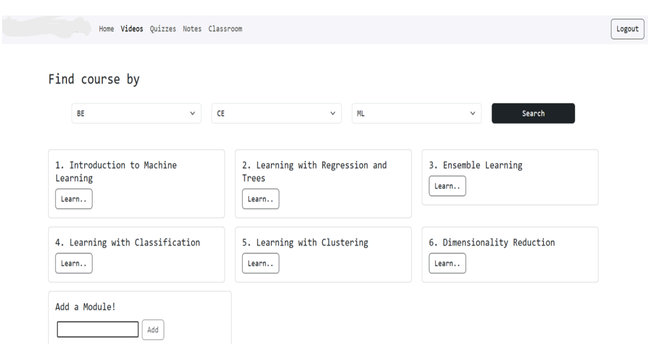

Fig. 4.3 - Resource and Video Uploading

SmartLearn allows educators to upload notes, study materials, and video content in various formats, including PDFs, PowerPoint presentations, and interactive multimedia, ensuring a rich learning experience. Teachers can organize resources into structured modules for easy student navigation. AI-generated transcripts, subtitles, and quizzes can supplement video content to reinforce learning outcomes. The platform supports interactive video features, live-streamed lectures, and recorded sessions, providing flexible learning options. A cloud-based repository stores all materials, enabling seamless retrieval and updates. Additionally, SmartLearn's smart recommendation system suggests supplementary resources based on student progress, fostering a personalized and interactive learning environment.

V. Future Direction

A. Improving ML Accuracy for Gesture Recognition

Currently, the hand gesture recognition model used in live lectures has limited accuracy. Future work will focus on enhancing the model's performance by experimenting with advanced algorithms such as CNNs, RNNs, and Transformer models. Incorporating more extensive and diverse training datasets will further improve the system's accuracy and reliability.

B. Enhancing UI/UX Design

To improve the user experience, SmartLearn's interface will be redesigned with a more intuitive, user-friendly layout. Focus will be placed on streamlining navigation, improving content organization, and ensuring accessibility across various devices. User feedback will be continuously incorporated to enhance the platform's functionality and visual appeal.

C. Preventing Cheating During Quizzes

To strengthen the platform's anti-cheating mechanisms, future updates will implement AI-powered proctoring with real-time monitoring, face recognition, and browser activity tracking. Additionally, enhanced copy-prevention techniques and randomized question sequences will be applied to minimize the chances of cheating during online assessments.

D. AI-Powered Personalized Learning Paths

SmartLearn will incorporate AI-based adaptive learning models to offer personalized content recommendations and dynamic learning paths. By analyzing student performance and engagement, the system can provide customized learning experiences, ensuring that students receive content aligned with their skill level and learning pace.

E. Mobile Application Development

To extend accessibility and improve engagement, a mobile application version of SmartLearn will be developed. This will ensure that students and teachers can access course materials, attend live lectures, and participate in quizzes anytime, anywhere.

F. Multilingual Support and Localization

To cater to a diverse student base, future versions of SmartLearn will include multilingual support and content localization. This will ensure that students from different regions can access content in their preferred language, enhancing inclusivity and learning outcomes.

VI. Conclusion

SmartLearn: Intelligent Learning Platform effectively integrates modern technologies to deliver a seamless and interactive digital learning environment. The platform empowers educators by enabling them to upload course content, create quizzes, track student performance, and conduct live lectures using advanced features like hand gesture recognition. AI-powered quiz generation, aligned with Bloom's Taxonomy, ensures comprehensive cognitive skill evaluation, while adaptive assessments personalize the learning experience based on student performance. Students benefit from easy access to structured course materials, interactive video content, and real-time feedback, fostering engagement and enhancing academic outcomes.

Moreover, SmartLearn incorporates robust security features, including secure login protocols and anti-cheating measures during quizzes, ensuring a fair and controlled learning environment. The platform's analytics dashboard offers valuable insights into student learning patterns, enabling educators to tailor their teaching strategies for improved outcomes. Real-time monitoring, progress tracking, and group project management features further enhance collaborative learning and ensure that students remain engaged and motivated.

While the current implementation of SmartLearn has demonstrated significant potential, future enhancements will focus on improving the accuracy of the hand gesture recognition model, enhancing UI/UX design, and integrating AI-powered personalized learning paths. Additionally, incorporating virtual labs, mobile application support, and multilingual content localization will expand the platform's capabilities, making it more accessible.

In conclusion, **SmartLearn** represents a significant advancement in digital education, offering an innovative, scalable, and data-driven solution that addresses the diverse needs of educators and students. By continually refining its features and incorporating emerging technologies, SmartLearn aims to create a transformative learning environment that fosters academic excellence, promotes engagement, and empowers learners to thrive in an increasingly digital world.

References

1. A. M. Al-Sharideh, T. H. Al-Rabadi, and M. S. Al-Hassan, "The impact of smart classrooms on the performance of students in higher education," *IEEE Access*, vol. 8, pp. 88833-88842, 2020. doi: 10.1109/ACCESS.2020.2992502.
2. A. H. S. S. J. O. Chik, "An implementation of a smart classroom system using IoT technology," *2018 IEEE 9th International Conference on Awareness Science and Technology (iCAST)*, pp. 249-254, 2018. doi: 10.1109/ICAwST.2018.8591401.
3. J. Wu, H. Yang, J. Yang, and Y. Zhang, "Development of smart classroom system based on Internet of Things," *2019 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pp. 58-63, 2019. doi: 10.1109/SmartIoT.2019.00017.
4. T. Zhang, L. Wang, and Y. Guo, "The application of artificial intelligence in smart classrooms," *2019 IEEE International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pp. 123-126, 2019. doi: 10.1109/ICAIBD.2019.8833020.
5. Y. Hu, Y. Wang, H. Wang, and Z. Jiang, "Research on the application of big data technology in smart classrooms," *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1-6, 2019. doi: 10.1109/BigComp.2019.8641985.
6. H. Zhang, "Exploration on the construction of smart classrooms based on 5G technology," *2020 IEEE International Conference on Computer Engineering and Technology (ICCET)*, pp. 176-180, 2020. doi: 10.1109/ICCET49364.2020.00048.
7. M. M. K. Alali, M. H. Ibrahim, and A. A. A. Al-Hussein, "A smart classroom management system using machine learning," *2020 IEEE International Conference on Cloud Computing and Artificial Intelligence (CCAI)*, pp. 162-167, 2020. doi: 10.1109/CCAI51114.2020.00038.

8. K. Lee and K. Kim, "Design and implementation of a smart classroom management system using machine learning," *2020 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 109-113, 2020. doi: 10.1109/ICCCS49541.2020.9053688.
9. P. K. Gupta, S. Y. Lee, and A. J. K. Y. Liu, "Enhancing learning through smart classrooms: A case study in engineering education," *2020 IEEE 22nd International Conference on Advanced Learning Technologies (ICALT)*, pp. 74-78, 2020. doi: 10.1109/ICALT49669.2020.00021.
10. S. M. Wibawa, D. A. Handayani, and H. N. Aini, “Development of an e-learning platform using a collaborative learning approach for higher education,” *Journal of Physics: Conference Series*, vol. 1193, no. 1, pp. 1–7, 2019.
11. R. Sharma and P. Sharma, “AI-based personalized learning: Enhancing education through adaptive learning technologies,” *International Journal of Artificial Intelligence and Education*, vol. 36, no. 2, pp. 123–135, 2022.
12. T. Zhang and A. Shah, “Gesture recognition using convolutional neural networks for interactive teaching,” *IEEE Access*, vol. 9, pp. 23456–23464, 2021.
13. M. A. Jadhav and S. S. Lathkar, “Automated question generation using NLP and machine learning techniques,” *International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 456–461, 2020.
14. Y. Xu, L. Wang, and H. Liu, “Improving online learning experiences through adaptive assessment and intelligent feedback,” *Journal of Educational Technology Systems*, vol. 49, no. 4, pp. 497–513, 2021.
15. G. V. Rao and K. B. Reddy, “Cloud-based learning management systems: A comprehensive review,” *International Journal of Cloud Computing and Education (IJCCE)*, vol. 15, no. 3, pp. 245–260, 2021.
16. H. Alqahtani and A. Rajkhan, “E-learning critical success factors during the COVID-19 pandemic: A comprehensive analysis,” *Education and Information Technologies*, vol. 26, pp. 7053–7075, 2021.
17. P. Mukherjee and R. Kumar, “AI-powered proctoring: Enhancing security in online assessments,” *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 4, pp. 129–140, 2022.
18. A. K. Yadav and S. Tiwari, “Virtual labs and simulations for enhancing engineering education: A case study,” *IEEE Global Engineering Education Conference (EDUCON)*, pp. 312–319, 2021.

b. Plagiarism Report

SmartLearn: Intelligent Learning Platform

ORIGINALITY REPORT

5%	1%	5%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

Progress review sheet 1 and 2

Inhouse/ Industry _Innovation/Research:

Class: D17 A/B/~~C~~

Sustainable Goal:

Project Evaluation Sheet 2024 - 25

Group No.: 52

Title of Project: SmartLearn : Intelligent Learning Platform (52)

Group Members: Shrinivas Ghumare, Pratik Waghmalkar, Mayank Wadhwani, Shubham Chelani

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life-long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	4	3	2	4	2	2	2	2	2	3	2	2	2	3	39

Comments:

M.D. Patil.

Name & Signature Reviewer1

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life-long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
4	4	3	2	4	2	2	2	2	1	3	2	2	2	2	37

Comments: Improve UI

Date: 1st March, 2025

Mrs. Manisha Mathur

Name & Signature Reviewer 2

Inhouse/ Industry _Innovation/Research:

Class: D17 A/B/~~C~~

Sustainable Goal:

Project Evaluation Sheet 2024 - 25

Group No.: 52

Title of Project: SmartLearn :- Intelligent Learning Platform (52)

Group Members: Shubham Chelani 04, Mayank Wadhwani 66, Shrinivas Ghumare (19), Pratik Waghmalkar (67)

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life-long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
5	5	5	3	4	2	2	2	2	2	2	3	3	2	2	44

Comments:

M.D. Patil

Name & Signature Reviewer1

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life-long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
5	5	5	3	4	2	2	2	2	2	3	3	3	2	2	45

Comments:

Date: 1st April,2025

Mrs. Manisha Mathur

Name & Signature Reviewer 2