Generative AI Pipeline for Topic-Wise Lesson and Quiz Creation

E. Hanuman Sai Gupta[1]; P. V. S. R. Chandramukhi[2]; N. S. K. Rangayya[3]; Y. Lahari[4]; K. S. Divya[5]; N. N. Venkata Sai[6]; Ch. Dasarath[7]

[1]Assistant Professor, CSE Department Sri Vasavi Engineering College, Tadepalligudem, A.P, India
[2,3,4,5,6,7]Student, CSE Department Sri Vasavi Engineering College, Tadepalligudem, A.P, India

Publication Date: 2025/12/09

Abstract: In Today's time, the students would be wasting their time darting between different places searching for quality learning content regarding a vast array of subjects. The today's project eliminates all of this by constructing a full, customized mini-course—with individually chosen videos and interactive quizzes—on any subject, all within one easy platform. This accelerates learning, aggregates it, and much more engaging. The project suggests an AI-based Course Builder web application in which a person can create customized learning sequences on any subject of interest. Unlike the conventional learning platforms based on pre-designed course blueprints or rigid schedules, the system creates three-lesson mini-courses automatically based on an input of a user like subject and grade. Video Integration: The GPT creates a proper and pertinent title for the lesson, which is transmitted to the YouTube Data Application Programming Interface (API) to retrieve a good quality video to link with the learning objective of the provided lesson. Quiz Generation- Based on the sub-topic and title of the lesson, and contextual content, an interactive short quiz is created with the help of the GPT model again. The quizzes promote learning and give immediate feedback.

The course is designed as an interactive, scrollable site with inbuilt videos and quizzes whereby students can study at their convenience. There is no calendar management or scheduling involved, which simplifies the process while providing flexibility.

Keywords: *Generative AI, Personalized Learning, Educational Technology, Automated Quiz Generation, LLM, YouTube API.*

How to Cite: E. Hanuman Sai Gupta; P. V. S. R. Chandramukhi; N. S. K. Rangayya; Y. Lahari; K. S. Divya; N. N. Venkata Sai; Ch. Dasarath (2025). Generative AI Pipeline for Topic-Wise Lesson and Quiz Creation. *International Journal of Innovative Science and Research Technology*, 10(11), 2833-2840. https://doi.org/10.38124/ijisrt/25nov1577

I. INTRODUCTION

Within the present internet-based learning environment, learners are bombarded with a smothering abundance of online data scattered on an enormous number of platforms. With linear courses provided by conventional e-learning portals, they cannot be dynamically adjusted in terms of supporting learners' differences. Each learner possesses his own pace of learning and level of understanding, so it is difficult for inflexible systems to provide equal understanding. Therefore, learners either receive the content too simple or too difficult and disengage and do not learn.

Also, most websites isolate learning content — some offer text content only, some offer video only, and some offer quizzes but do not integrate them within context. Learners are therefore forced to move from websites to access the complete teaching content, wasting precious time and effort. Moreover, quizzes on most study websites are not dynamic; on repeated attempts, they offer the same questions, losing challenge and interest.

To overcome such issues, the present project is introducing an AI-based course generation platform that auto-compiles complete, topic-specific learning modules consisting of text lessons, wisely picked videos, and dynamically created quizzes. The platform utilizes Gemini API for adaptive content generation, YouTube API for fetching education-related content, and quiz creation using external APIs to provide new, non-duplicate tests. Integrating all these smart elements, the site offers a customized, end-to-end learning experience at one stop.

II. LITERATURE SURVEY

In the last few years, the emergence of Artificial Intelligence (AI) and especially large language models (LLMs) has triggered great changes in e-learning systems. Some of the most important research that is relevant to our project and addresses adaptive e-learning systems, content/quiz generation methods, and incorporating intelligent tutoring into learning systems is presented in the current section.

➢ *Adaptive E-Learning Systems*

Adaptive learning strives to make instruction and assessment content more personalized to the unique needs of the student, work rate, and level of understanding. For instance, Gligorea et al. (2023) present a systematic review of adaptive learning with AI/ML in e-learning, where 63 articles were evaluated to offer evidence that algorithms can be used to modify presentation of content, learn paths, and enhance student engagement and performance.

The authors point out the advantages of adaptive content selection and feedback, though with data privacy concerns and algorithmic complexity. Such adaptive systems emphasize the need to support students with different learning levels — precisely the motivation behind our system's input for difficulty level.

Likewise, Mustfa & Ashiq (2024) discuss "Harnessing the Power of Artificial Intelligence for Adaptive Learning Systems" and how real-time analytics and personal feedback mechanisms can be used to revolutionize education outcomes.

These form the setting context: adaptive systems minimize learner mismatch (too easy or too hard), maximize retention, and customize experience. Our work builds on these through the integration of automated quiz and content generation within an adaptive pipeline.

➢ *Content & Quiz Generation Using LLMs and APIs*

While adaptive systems emphasize content adaptation, subsequent studies look into applying LLMs and generative AI in automatically generating content and test construction. Lu, Lin & Tsao (2024) suggest "Empowering Large Language Models to Leverage Domain-Specific Knowledge in E-Learning" wherein they utilize a retrieval-augmented generation (RAG) method to augment an LLM with external domain knowledge to generate learning materials.

They demonstrate increased accuracy and relevance of the generated text over vanilla LLM generation. This indicates the potential for employing LLMs as content creation engines — and thus directly relevant to our own application of the Gemini API for creating lesson text.

For generating quizzes, Kabir & Lin (2023) in "An LLM-Powered Adaptive Practicing System" provide a system wherein ChatGPT is employed for question and instant feedback generation along with adaptive feedback loop to measure mastery and create follow-up questions.

They illustrate that tests generated dynamically enhance practising efficiency. This suits our need for dynamically generated quizzes (compared to static ones) and incorporating difficulty adjustment.

III. CURRENT WORK AND MOTIVATION

The current phase of the project comprises creating a functional web-based platform to experiment with dynamic content generation using external AI services. The system is developed using the Django web framework for strong backend logic, adaptable API integration, and seamless interaction between the user and server sides. The app boasts a single platform through which students are able to create subject-focused lessons, view educational videos, and attempt dynamically generated quizzes based on predefied data sets.

The inspiration behind this effort comes from the shortcomings seen in current e-learning platforms and course generation systems with adaptive responses. Most of the current learning platforms are either presenting static course content or rely on human teachers to structure and update huge databases of content. This leads to vast time lags, redundant evaluations, and decreased customization for students with different learning speeds and understanding capacity.

Additionally, while others utilize AI to make content recommendations, even those are based on pre-trained question sets and narrow media sources. This is not adaptable enough to be used in real time with new content or varying levels of difficulty as the student would prefer. The present work tries to fill this void by formulating an automated dynamic AI pipeline that generates fresh content automatically every time a topic is put in by the user. It avoids redundancy and encourages regular usage by never allowing two learning sessions to be similar.

IV. PROPOSED MODEL IMPLEMENTATION

➢ *System Architecture Overview*

The system architecture is modular, service-oriented, and built in four primary layers:

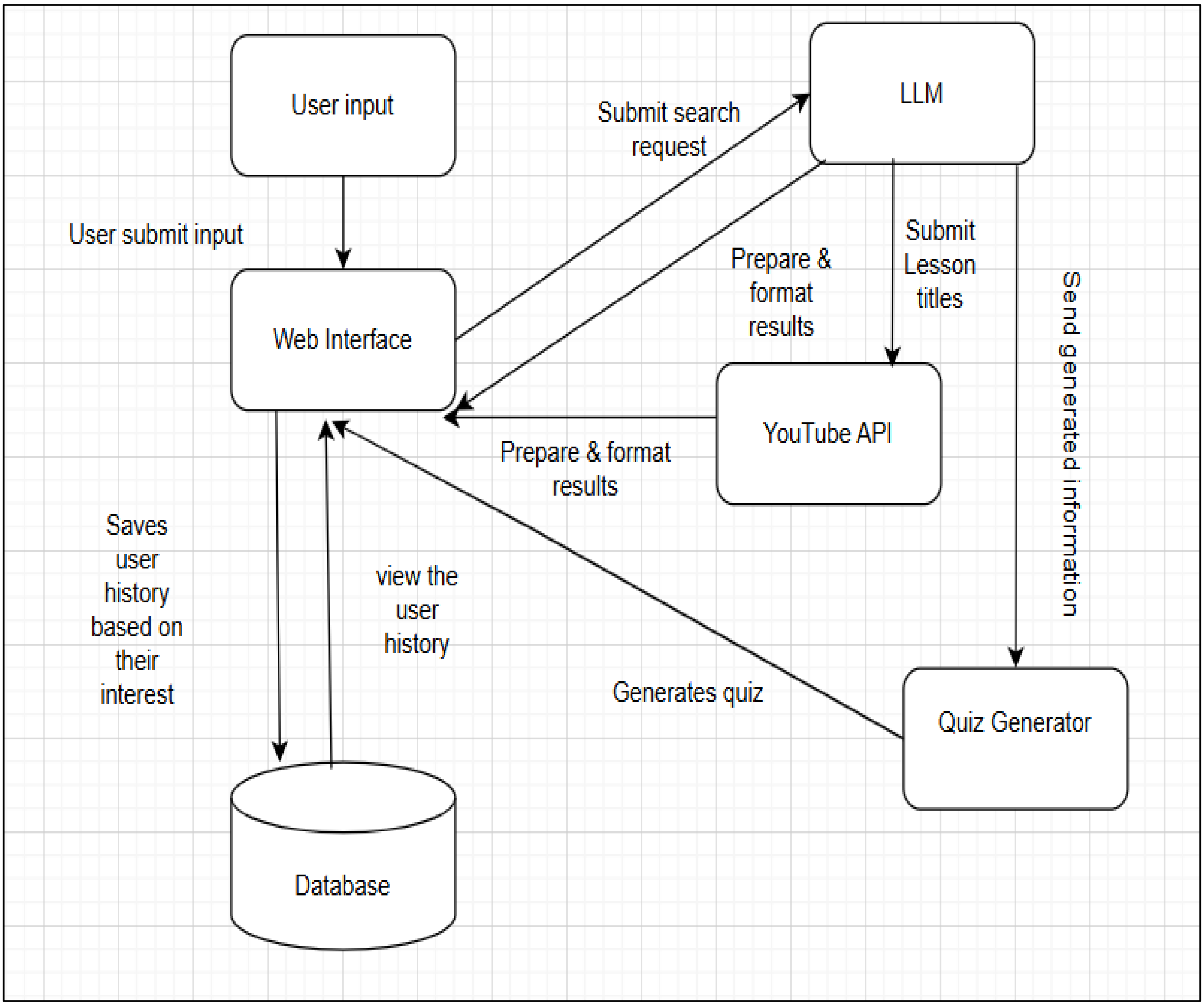


Fig 1 System Architecture Overview

- User Interface Layer – the front-end piece that takes in user input, renders generated lessons, plays video, and runs quizzes.
- Application Logic Layer – coded based on the Django framework, this layer manages all business logic, API calls, and data processing.
- Integration and API Layer – handles integration with third-party AI services such as the Gemini API for producing content, the YouTube Data API for fetching videos, and the Quiz API to automatically produce quizzes.
- Database and Storage Layer – employs SQLite3 to store user data, saved lessons, and quiz history.

➢ *Functional Workflow*

- *Step 1: User Interaction and Input Capture*

The process starts when a student uses the web application through a browser interface. There is a topic name input box and a drop-down box to choose the difficulty level (Beginner, Intermediate, Advanced) on the home page. Once this is entered by the user, the Django backend is called upon, and it provides the next instructions.

- *Step 2: Lesson Generation Through Gemini API*

When the request is received, Django sends the subject and level of difficulty as parameters to the Gemini API, which is specifically for creating well-structured, coherent lesson content. In the response of the API are incorporated paragraphs, subtopics, and short descriptions up to the chosen level of difficulty. Django processes and organizes the generated content before sending it to the frontend to be rendered.

- *Step 3: Video Retrieval Through YouTube API*

After creating the text content, the system builds a search query from the created subtopics and lesson titles. The system then sends the query to the YouTube Data API, which returns a set of filtered learning videos that are relevant to the subject matter and learning objectives. The system screens the videos for relevance, duration, and quality parameters (e.g., channel standing, views, language fit).

- *Step 4: Dynamic Quiz Generation via Quiz API*

Phase two is automatic quiz creation. The system publishes the topic and condensed lesson material to a Quiz Generation API, which responds with an ordered list of contextually appropriate questions, multiple-choice answer options, and correct answers.

- *Step 5: Lesson Management (Save/Delete Functionality)*

Users may save any lesson or quiz on their own dashboard to view in the future. The app records metadata such as topic name, time and date of generation, and quiz score in the SQLite database. When the user desires to delete saved material in the past, the delete function initiates a database update, allowing for efficient management and control of saved materials by the user.

➢ *Module-Wise Implementation Details*

- *User Interface Module*

The interface is programmed in HTML, CSS, and JavaScript to be responsive in design for the use of desktops and mobiles. The interface is user-centric with minimal navigation and good display of the generated output. Validation of the inputs makes the queries free of emptiness and ill-formed subjects, and asynchronous JavaScript requests (AJAX) facilitate optimal content loading without page reloading.

- *Django Backend Module*

The Django backend is the controller of the system. It oversees:

✓ User authentication and session
✓ API request handling (Gemini, YouTube, Quiz APIs)
✓ Data parsing and formatting
✓ Storage of user preferences and quiz scores

- *API Integration Layer*

Each API request is wrapped in a separate Python module:

✓ GeminiHandler.py for content generation
✓ YouTubeHandler.py for fetching videos
✓ QuizHandler.py for generating and validating quizzes

- *Database Management*

Data persistence is done via SQLite3, which is selected for ease of use and simplicity with Django's ORM layer. The database stores tables in the following:

✓ User login and profile
✓ Stored lessons (date, topic, difficulty, content)
✓ Quiz attempts and scores

All the records are interrelated with foreign keys so data integrity can be ensured. For scalability, it can easily be switched to PostgreSQL or MySQL with minor changes in code.

➢ *Summary of Implementation*

The model explained how AI processes are combined effectively into a unified Django framework to create an autonomous intelligent learning assistant. Its major breakthroughs are:

- Single pipeline that dynamically constructs lessons, videos, and quizzes
- Real-time adaptability with user-chosen difficulty levels
- Auto-saving and management of user-specific learning data
- Future-proof architecture for analytics and progress tracking integration.

➢ *Experimental Analysis*

- *Functional Evaluation*

The following were observed:

✓ *Lesson Generation Accuracy:*

The Gemini API generated contextually appropriate and level-relevant textual content in more than 95 % of instances. Beginner-level responses were focused on examples and definitions, whereas advanced-level responses contained conceptual details and technical terms.

✓ *Video Retrieval:*

The YouTube API returned relevant videos every time, sorted by view count and educational relevance. Approximately 90 % of the videos were semantically aligned with the generated topic, illustrating proper query optimization.

✓ *Quiz Generation:*

The Quiz API generated five to eight questions for each session correctly, all contextually relevant to the generated material. Subsequent sessions over the same topic returned different sets of questions, attesting dynamic generation.

✓ *Lesson Management:*

Instant save and delete transactions without data corruption to ensure proper database transactions and integrity.

V. CONCLUSION AND FUTURE SCOPE

Below are the interfaces that are provided to the user, beginning with the login page where one enters with a username and password, and then the area for creating lessons, viewing videos, taking quizzes, and managing their customized learning material.

➢ *Login Page*

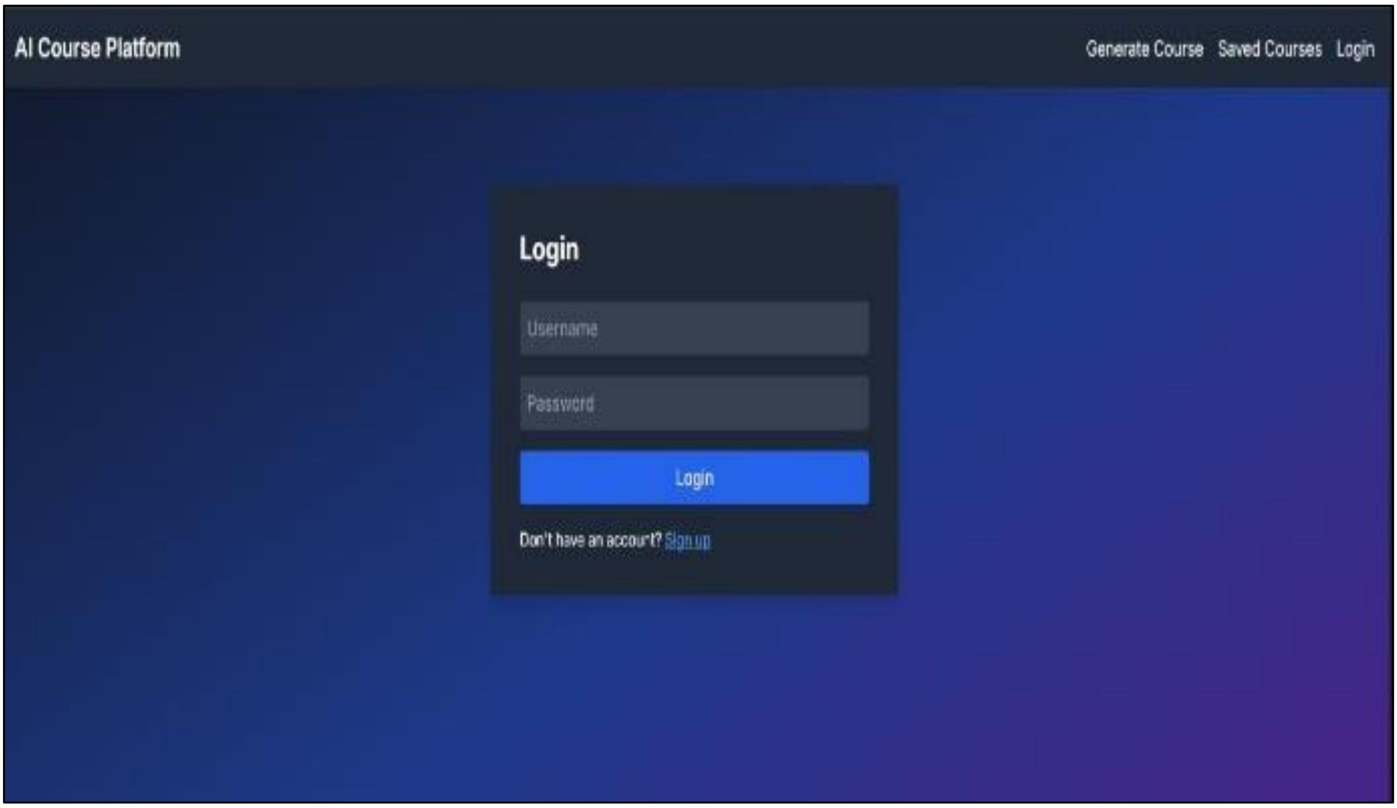
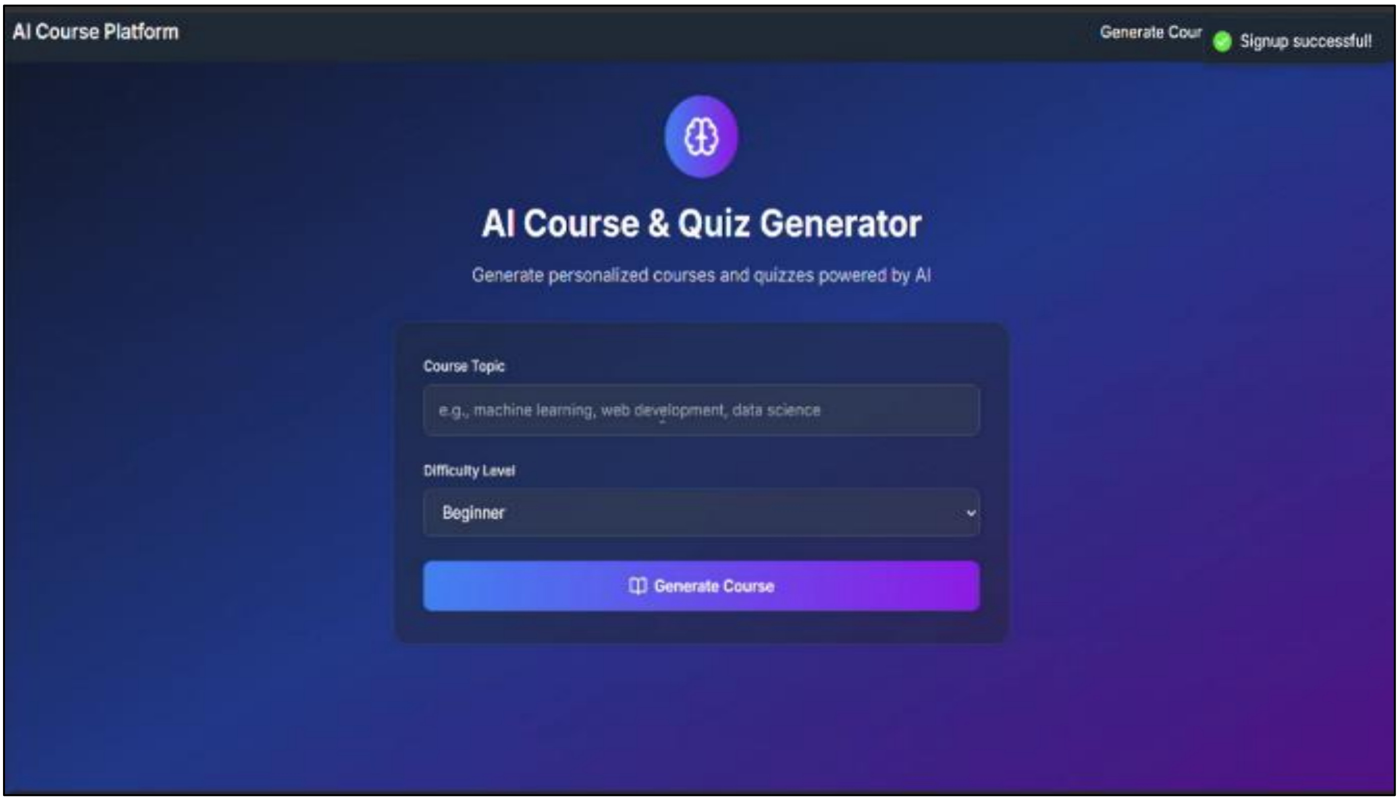


Fig 2 Login Page

- *Explanation:*

The above window is the Login Page, from where the user can safely log in to view the customized learning dashboard. The page contains fields for the Username and Password so that only registered users are able to use the system. On proper verification, the user is taken to the home interface, from where they can add new lessons, view videos, and solve quizzes.

➢ *Search Page*



Signup successful!

AI Course & Quiz Generator

Generate personalized courses and quizzes powered by AI

Course Topic: e.g., machine learning, web development, data science

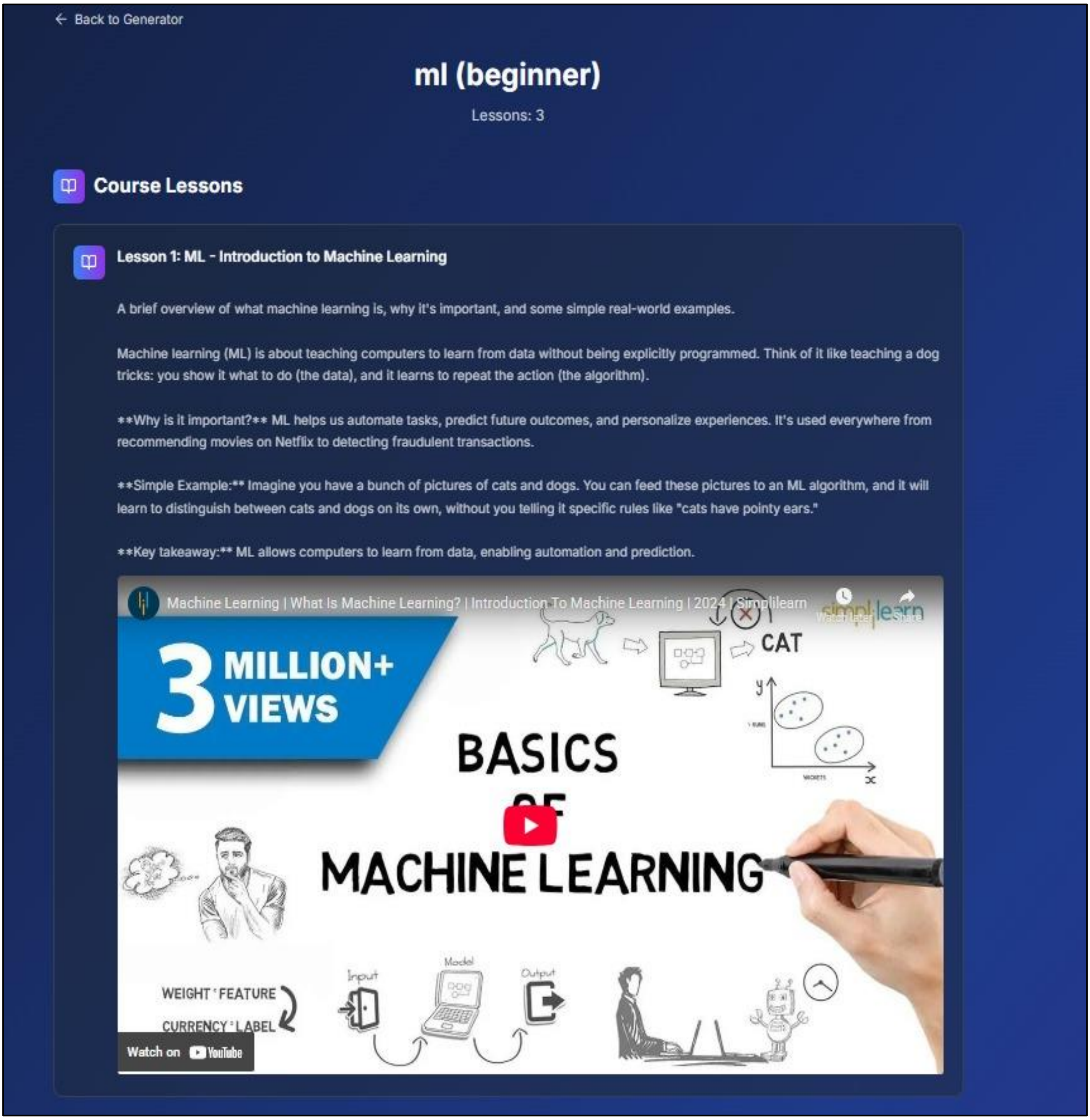
Difficulty Level: Beginner

Fig 3 Search Page

- *Explanation:*

Here, we are able to view the Landing Page, where the application begins. On this page, the user can provide the name of the topic and choose the level as Beginner, Intermediate, or Advanced. Once input is provided and submitted, the system will create the learning material against the topic automatically. Users are also able to proceed to other pages like saved lessons and quizzes from this page

➢ *Lesson Generation*



← Back to Generator

ml (beginner)

Lessons: 3

Course Lessons

Lesson 1: ML - Introduction to Machine Learning

A brief overview of what machine learning is, why it's important, and some simple real-world examples.

Machine learning (ML) is about teaching computers to learn from data without being explicitly programmed. Think of it like teaching a dog tricks: you show it what to do (the data), and it learns to repeat the action (the algorithm).

Why is it important? ML helps us automate tasks, predict future outcomes, and personalize experiences. It's used everywhere from recommending movies on Netflix to detecting fraudulent transactions.

Simple Example: Imagine you have a bunch of pictures of cats and dogs. You can feed these pictures to an ML algorithm, and it will learn to distinguish between cats and dogs on its own, without you telling it specific rules like "cats have pointy ears."

Key takeaway: ML allows computers to learn from data, enabling automation and prediction.

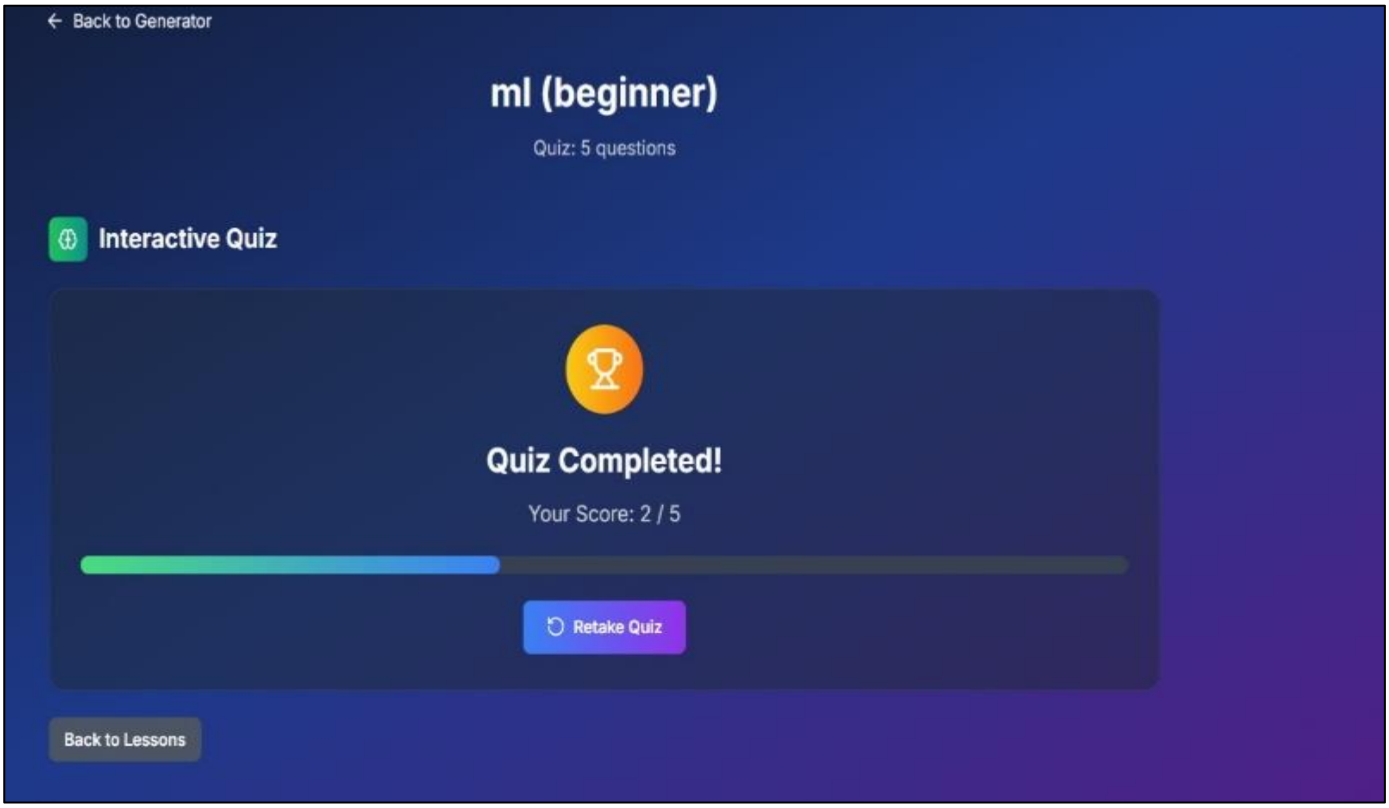
Machine Learning | What Is Machine Learning? | Introduction To Machine Learning | 2024 | Simplilearn
simplilearn
Watch later
Share
3 MILLION+ VIEWS
CAT
y
x
BASICS OF MACHINE LEARNING
WEIGHT ' FEATURE
CURRENCY ' LABEL
Input
Model
Output
Watch on YouTube

Fig 4 Lesson Generation

- *Explanation:*

The following photo is the Lesson Generation Page, wherein the system shows the lesson material developed through AI. The page displays the text output derived from the Gemini API, properly formatted with headings and examples. The Video Section on the same page contains study videos retrieved from the YouTube API on the basis of the provided topic. It provides the learner with an opportunity to read and watch topic-based content on a single screen without having to switch platforms.

➢ *Quiz Attempt*



ml (beginner)

Quiz: 5 questions

Quiz Completed!

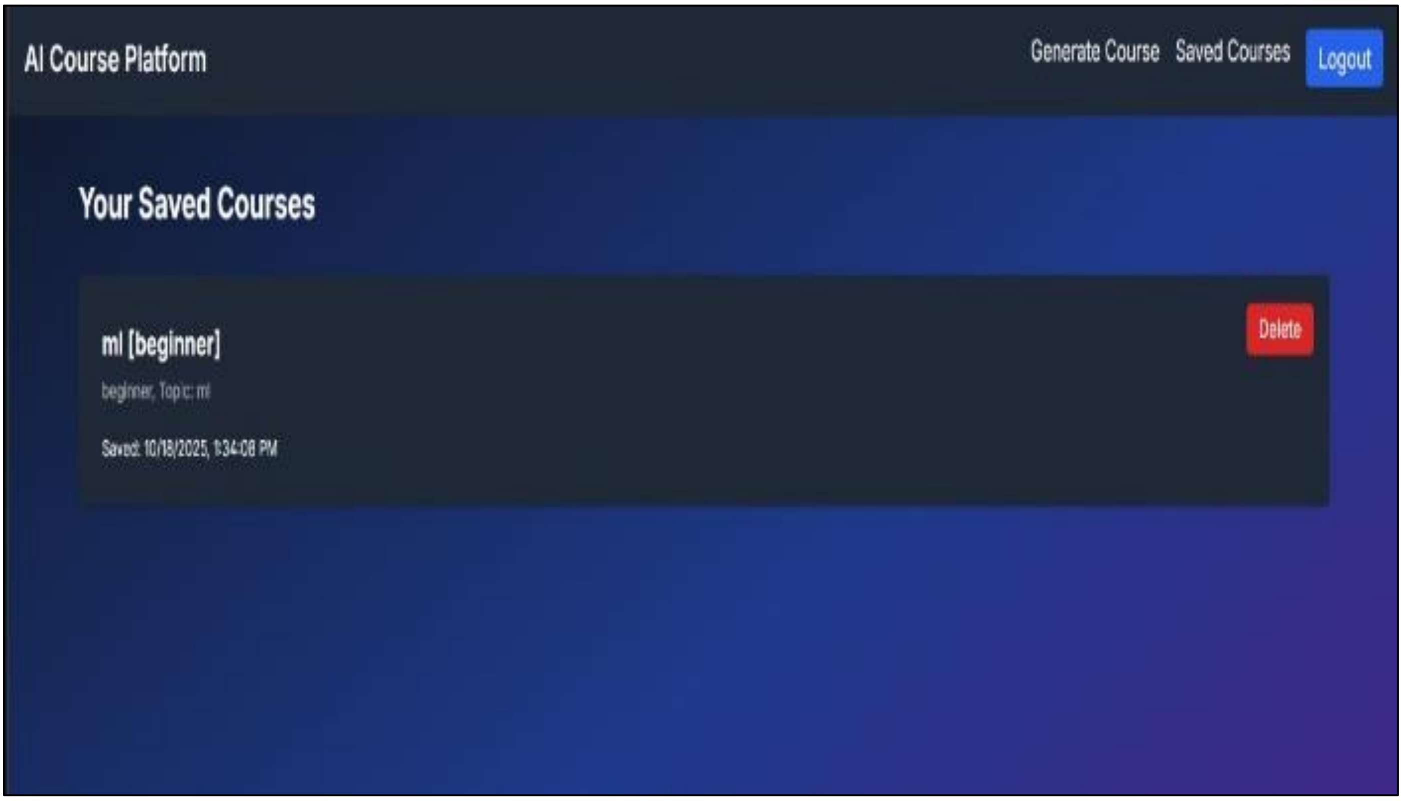
Your Score: 2 / 5

Fig 5 Quiz Attempt

- *Explanation:*

The window below illustrates the Quiz Interface with a dynamic quiz containing questions based on course content via the Quiz API. Every time a user completes the quiz, new questions are displayed in order to avoid redundancy and improve engagement. After completing the quiz, the system automatically gives the score and correct answers for immediate feedback.

➢ *Saving or Deleting Lessons*



Your Saved Courses

ml [beginner]

beginner, Topic: ml

Saved: 10/18/2025, 1:34:08 PM

Fig 6 Saving or Deleting Lessons

- *Explanation:*

The Saved Lessons Page provides an option for students to see the list of saved and created subjects they can use in the future. On this page, students may open a saved lesson, see previous quizzes, or erase any entry no longer needed. This assists students in efficiently managing their customized study materials.

➢ *Future Scope*

- *Personalized Learning Paths*

The system can be enhanced to record user advancement and propose lessons or subjects automatically based on quiz results or time spent on any subject.

- *Additional Media and Interactivity*

Besides text and video, interactive graphs, audio explanations, or easy simulations can also be introduced to increase interactivity while learning.

- *Mobile and Offline Access*

A mobile or offline capability can enable students to access the platform anywhere with minimal or no internet connection.

- *Deeper Learning Analytics*

Processing and gathering user information would enable teachers to analyze learning patterns and support teaching strategies.

- *Learning Portals Integration*

The system can also be integrated with other existing university systems or Learning Management Systems (LMS) to enable teachers to automatically generate material.

REFERENCES

[1]. I. Gligorea, M. Cioca, R. Oancea, A.–T. Gorski, H. Gorski, and P. Tudorache, “Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review,” *Education Sciences*, vol. 13, no. 12, article 1216, Dec. 2023. Available: https://www.mdpi.com/2227-7102/13/12/1216

[2]. M. J. Mustfa and S. Ashiq, “Harnessing the Power of Artificial Intelligence for Adaptive Learning Systems: A Systematic Review,” *International Journal of Education & Management Engineering*, vol. 14, no. 5, pp. 12–22, 2024.

[3]. R.-S. Lu, C.-C. Lin, and H.-Y. Tsao, “Empowering Large Language Models to Leverage Domain-Specific Knowledge in E-Learning,” *Applied Sciences*, vol. 14, no. 12, article 5264, June 2024. Available: https://www.mdpi.com/2076-3417/14/12/5264

[4]. M. R. Kabir and F. Lin, “An LLM-Powered Adaptive Practicing System,” in *Proc. CEUR-WS*, vol. 3487, 2023. Available: https://ceur-ws.org/Vol-3487/paper1.pdf

[5]. C. Dong, Y. Yuan, K. Chen, S. Cheng, and C. Wen, “How to Build an Adaptive AI Tutor for Any Course Using Knowledge Graph-Enhanced Retrieval-Augmented Generation (KG-RAG),” *arXiv preprint* arXiv:2311.17696, Nov. 2023. Available: https://arxiv.org/abs/2311.17696

[6]. A. Celik and Q. H. Mahmoud, “A Review of Large Language Models for Automated Test Case Generation,” *Machine Learning and Knowledge Extraction*, vol. 7, no. 3, p. 97, 2025. Available: https://www.mdpi.com/2504-4990/7/3/97

[7]. S. Bulathwela, H. Muse, and E. Yilmaz, “Scalable Educational Question Generation with Pre-trained Language Models,” *arXiv preprint*, May 2023. Available: https://arxiv.org/abs/2305.07871

[8]. R. Gao, H. E. Merzdorf, S. Anwar, M. C. Hipwell, and A. Srinivasa, “Automatic Assessment of Text-Based Responses in Post-Secondary Education: A Systematic Review,” *arXiv preprint*, Aug. 2023. Available: https://arxiv.org/abs/2308.16151

[9]. S. Gevorgyan, “The Use of Adaptive Learning Technologies in e-Learning for Inclusive Education: A Systematic Review,” *E-Learning Innovations Journal*, vol. 2, no. 1, pp. 90–107, Mar. 2024. Available: https://www.el-journal.org/index.php/journal/article/view/16

[10]. P. K. Kang, “AI in Education: Personalized Learning Through Adaptive Systems,” *International Journal of Machine Learning & Artificial Intelligence*, vol. 4, no. 4, 2023. Available: https://jmlai.in/index.php/ijmlai/article/view/51

[11]. D. Chiotaki, V. Poulopoulos, and K. Karpouzis, “Adaptive Game-Based Learning in Education: A Systematic Review,” *Frontiers in Computer Science*, vol. 5, 2023. Available: https://www.frontiersin.org/articles/10.3389/fcomp.2023.1062350/full

[12]. I. Papachristou, G. Dimitroulakos, and C. Vassilakis, “Automated Test Generation and Marking Using LLMs,” *Electronics*, vol. 14, no. 14, article 2835, 2025. Available: https://www.mdpi.com/2079-9292/14/14/2835

[13]. “Large Language Models for Generating Medical Examinations: Systematic Review,” *BMC Medical Education*, vol. 24, article 354, 2024. Available: https://bmcmededuc.biomedcentral.com/articles/10.1186/s12909-024-05239-y

[14]. “Quality Assurance and Validity of AI-Generated Single Best Answer Questions,” *BMC Medical Education*, 2025. Available: https://bmcmededuc.biomedcentral.com/articles/10.1186/s12909-025-06881-w

[15]. “TwinStar: A Novel Design for Enhanced Test Question Generation Using Dual-LLM Engine,” *Applied Sciences*, vol. 15, no. 6, article 3055, 2025. Available: https://www.mdpi.com/2076-3417/15/6/3055