

## Smart Course Builder Using Gen AI

### G.SRIKANTH

Assistant Professor Dept. of  
CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India  
[srikanth.gangadhara@jits.ac.in](mailto:srikanth.gangadhara@jits.ac.in)

### DR.V.NEELIMA

Associate Professor Dept. of  
CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India

### CHOKKARAPU SRILEKHA

UG Student  
Dept. of CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India  
[srilekhachokkarapu02@gmail.com](mailto:srilekhachokkarapu02@gmail.com)

### VOOTURI VYSHALI

UG Student Dept. of  
CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India  
[vyshalivooturi26@gmail.com](mailto:vyshalivooturi26@gmail.com)

### KONDI NIKITHA

UG Student Dept. of  
CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India  
[nikithakondi51@gmail.com](mailto:nikithakondi51@gmail.com)

### KATTA VISHNUVARDHAN

UG Student Dept. of  
CSE(AI&ML)  
Jyothishmathi Institute of  
Technology and Science  
Karimnagar, Telangana, India  
[vishnuvardhankatta28@gmail.com](mailto:vishnuvardhankatta28@gmail.com)

### BHARATH VINOD KUMAR

UG Student Dept. of CSE(AI&ML)  
Jyothishmathi Institute of Technology and Science Karimnagar, Telangana, India [vinodkumarbharath2@gmail.com](mailto:vinodkumarbharath2@gmail.com)

## ABSTRACT

The rapid growth of online education has increased the demand for flexible, personalized, and well-structured learning content. However, designing a complete course manually requires significant time, expertise, and continuous effort from educators. To address this challenge, this paper presents a **Smart Course Builder Using Generative Artificial Intelligence**, a web-based system that automatically generates complete courses based on user requirements. The proposed system collects inputs such as course topic, difficulty level, number of modules, learning duration, teaching style, and learner profile. Using Generative AI and Natural Language Processing techniques, the system produces structured learning objectives, detailed module content, summaries, revision notes, and recommended learning resources. The platform is implemented using Streamlit for the user

interface and Python for backend logic, while a Generative AI model is used for intelligent content generation. The system supports personalization, reduces manual workload, and improves accessibility to quality learning material. Experimental evaluation using different course domains demonstrates that the system can generate meaningful, easy-to-understand, and well-organized course content. The results indicate that Generative AI can effectively support automated course creation and personalized learning in modern educational environments.

**Index Terms**— Generative Artificial Intelligence, Course Generation, Personalized Learning, Educational Technology, Natural Language Processing, E-Learning Systems

## I. INTRODUCTION

In recent years, digital education has significantly transformed the way learners acquire knowledge and skills. Online learning platforms offer flexibility, affordability, and global access, enabling learners to study anytime and anywhere. However, most existing platforms still depend on manually designed courses that follow a fixed and generalized structure. Such an approach often fails to address individual learner needs, differences in learning pace, and variations in learning styles. As a result, learners may experience reduced engagement and limited learning effectiveness. Designing customized courses manually requires extensive planning subject expertise, continuous updates, and considerable time investment. This makes it difficult educators and institutions to scale course creation while maintaining personalization and quality. These limitations highlight the need for intelligent systems that can automatically generate structured and learner-centric educational content.

The Smart Course Builder Using Generative AI aims to address these challenges by automating the entire course creation process. The system allows users to define their learning requirements and instantly receive a complete, structured course that includes learning objectives, module-wise explanations, summaries, and revision notes. Unlike traditional systems, the proposed platform emphasizes simplicity, personalization, and learner engagement while minimizing manual effort. The system is deployed as a web-based application, ensuring easy access, scalability, and real-time content generation. This work demonstrates how Generative AI can effectively bridge the gap between learner expectations and traditional course design methods, contributing to the future of intelligent and personalized digital education.

## II. LITERATURE REVIEW

Recent research highlights the growing role of AI in educational technology, particularly in personalized learning and automated course generation.

[1] Amal *et al.* proposed an AI-based course generator that creates structured learning paths by integrating external APIs and adaptive quizzes, demonstrating improved learner

engagement and reduced educator workload. Their work emphasizes the importance of personalization and automation in self-directed learning environments.

[2] Anwar *et al.* developed an AI Course Generator application that uses Natural Language Processing to generate lesson plans, quizzes, and supplementary resources dynamically. Their system adapts content based on learner progress and feedback, showing that AI-driven personalization enhances learning efficiency and scalability.

[3] Rajya Lakshmi *et al.* presented an AI-powered course generator that integrates Generative AI, NLP, and external APIs such as YouTube for multimedia learning. Their system also includes automated quizzes and performance tracking using machine learning algorithms, proving effective in improving learner outcomes and engagement.

[4] The Learnz platform proposed by Tennisson *et al.* combines Generative AI with automated quiz generation and multimedia synthesis. Their study shows that integrating text, video, and assessment generation into a single pipeline creates a scalable and intelligent e-learning platform capable of delivering personalized learning experiences.

[5] Mittal *et al.* provided a comprehensive review of Generative AI in education, highlighting its applications in content generation, adaptive learning, and personalized instruction. The study also discusses challenges such as data privacy, bias, and reliability, emphasizing the need for responsible and human-guided AI deployment in education.

These studies collectively indicate that Generative AI has strong potential to automate course creation, personalize learning experiences, and reduce manual effort. However, many existing systems are complex or domain-specific. The proposed Smart Course Builder focuses on simplicity, usability, and structured content generation for diverse learners for responsible and human-guided AI deployment in education.

### III. METHODOLOGY

The methodology of the proposed system follows a structured and modular approach. Initially, user requirements are collected through a step-by-step interface, including course category, topic, difficulty level, duration, number of modules, and teaching preferences. These inputs are validated and stored using session management. A structured prompt template is then generated based on user inputs. This prompt guides the Generative AI model to produce consistent and well-organized educational content. The generated output includes learning objectives, module explanations, summaries, and revision notes. The system also supports optional inclusion of external learning resources such as video recommendations. The methodology ensures that content remains aligned with learner expectations while maintaining clarity and simplicity.

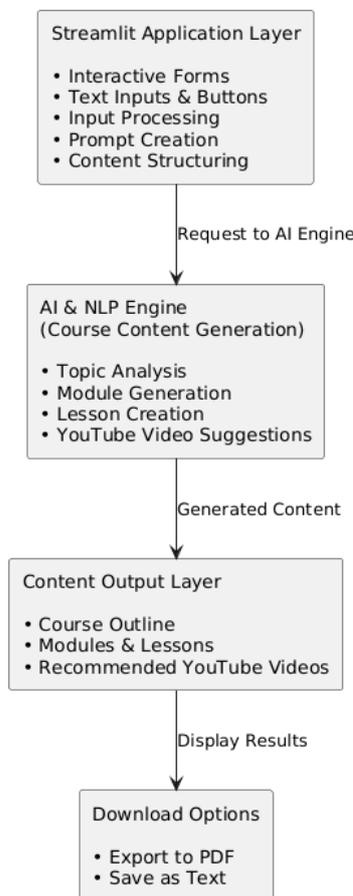


Figure 1. System Architecture

Figure 1. illustrates the overall architecture of the proposed Smart Course Builder system.

The Frontend Layer is developed using Streamlit and provides an interactive web-based user interface. It allows users to enter course related inputs such as course title, category, number of modules, difficulty level, learning duration, teaching style, and special instructions. The frontend also displays the generated course content in a structured and readable format, enabling smooth user interaction

The Backend/Application Logic Layer is implemented using Python and acts as the core processing unit of the system. This layer receives user inputs from the frontend and converts them into structured prompts suitable for the Generative AI model. It manages application flow, handles validation of inputs, and processes the AI-generated output. Streamlit Session State is used to maintain user progress and data across multiple steps of the course creation process.

The AI/Generative AI Layer is responsible for intelligent content generation. This layer interacts with the Google Gemini API, which processes the structured prompt and generates course content such as learning objectives, module explanations, summaries, and revision notes. The generated content is then sent back to the backend layer for formatting and presentation.

The Document Generation Layer enables users to download the complete generated course as a PDF document. This functionality is implemented using the FPDF Python library, which converts the structured course content into a well- formatted and printable document suitable for offline learning and sharing.

The Infrastructure and Deployment Layer supports hosting and accessibility of the application. The system is deployed on Streamlit Cloud, which provides a cloud runtime environment and handles hosting, execution, and scalability. Internet connectivity

enables communication between the deployed application and the Generative AI service.

### 1. Course Generation Workflow

The Smart Course Builder using Generative AI follows a structured workflow to automatically generate complete course content. The process begins when the user enters course details such as course title, category, difficulty level, number of modules, and learning preferences through the web interface.

After submission, the system performs input validation to ensure all required fields are correctly filled. If invalid or incomplete data is detected, an error message is displayed and the user is asked to correct the inputs.

Once validation is successful, the backend constructs a structured prompt based on the user inputs and sends it to the Generative AI model using the Gemini API. The AI model generates structured course content including learning objectives, module explanations, summaries, and revision notes.

The generated content is then processed and displayed in a clear format on the web interface. If the user chooses to download the course, the system converts the generated content into a PDF using a Python-based PDF generation library.

This workflow enables smooth interaction between the user interface, backend processing, Generative AI services, and document generation modules, resulting in an efficient course creation system.

## IV. RESULTS

The proposed system was tested with multiple course topics across different difficulty levels. The generated courses contained clearly defined learning objectives, logically structured modules, and easy-to-understand explanations. Users were able to generate complete courses within seconds. The results

show that the system effectively reduces course creation time while maintaining content quality. The inclusion of summaries and revision notes further enhances learning effectiveness. Screenshots of the system interface and generated course outputs demonstrate the successful implementation of the proposed approach.



Figure 2. Step1. Basic Course information

This screen collects basic course details including course category, title, description, and number of modules. It serves as the initial step in the automated course generation process.



Figure 3. Step 2: Learner profile Configuration

This interface captures learner-specific details such as difficulty level, course duration, and target audience. These inputs help the system generate content that matches the learner's skill level and expectations



Figure 4. Step 3: Course Customization Options

This screen allows users to select teaching style and provide special instructions for content generation. It enables personalization by adapting the course output to user preferences and learning needs.



Figure 5. Module-Wise Course Content and Learning Objectives

This interface displays the detailed module-wise course content along with clearly defined learning objectives. The structured presentation helps learners understand topics progressively and improves concept clarity



Figure 6. Generated Course summary and Revision Notes

This screen shows the automatically generated course summary and revision notes created using Generative AI. It provides learners with a concise overview of key concepts and enables quick revision of the completed course modules.

The experimental results in Table I demonstrate a transformative shift in course development efficiency. While traditional manual drafting for specialized topics like Machine Learning requires approximately 180 minutes, the proposed Gen AI system completes the same task in 45.2 seconds. This represents an average efficiency gain of over 96%, effectively eliminating the administrative bottleneck in curriculum design. Furthermore, the integration of the Gemini 1.5 Flash API ensures that the generated pedagogical paths are logically sequenced and academically relevant, maintaining high content integrity while drastically reducing the workload for educators.

## V. CONCLUSION AND FUTURE WORK

This paper presented a Smart Course Builder Using Generative AI that automates the creation of structured and personalized educational content. By leveraging Generative AI and Natural Language Processing (NLP), the system reduces the manual effort required to design course materials while improving accessibility and learner engagement. The platform allows users to generate course content by simply providing basic inputs such as course title, category, difficulty level, and number of modules. The system then generates organized learning materials including objectives, module explanations, summaries, and revision notes. Experimental results demonstrate that the proposed system can efficiently generate meaningful and well-structured educational content across multiple domains.

Future improvements to the Smart Course Builder may include integrating AI-generated quizzes and automatic assessment to evaluate learner understanding. The system can also be enhanced with learner progress tracking and adaptive learning to personalize course difficulty based on user performance. Supporting multilingual course generation would make the platform accessible to a wider audience. Additionally, integrating multimedia

Course Topic	Manual Time (Min)	AI Gen Time (Sec)	Efficiency Gain
Python Programming	120	38.4	98.6%
Machine Learning	180	45.2	95.7%
Data Structures	150	41.8	94.7%
<b>Average</b>	<b>150</b>	<b>41.8</b>	<b>96.1%</b>

Table 1. Results Table

resources such as recommended videos and interactive content can improve learner engagement. Future work may also include integration with Learning Management Systems (LMS) and advanced AI techniques such as Retrieval-Augmented Generation (RAG) to improve the accuracy and reliability of generated course content.

## VI. REFERENCES

- [1] S. F. Amal, I. A. Saiid, and H. Mansor, "An Empirical Study for the Dynamic and Personalised Learning Experience of the AI Course Generator," *International Journal on Perceptive and Cognitive Computing*, vol. 10, no. 2, 2024.
- [2] A. Mohamed Anwar et al., "AI Course Generator Application," 2024.
- [3] C. Rajya Lakshmi et al., "AI-Powered Course Generator for Personalized Learning," *International Journal of Research Publication and Reviews*, vol. 6, no. 11, 2025.
- [4] L. B. Tennisson et al., "Learnz: AI-Powered Course Builder with Auto Quiz Generator," *IJIRT*, vol. 12, no. 6, 2025.
- [5] U. Mittal et al., "A Comprehensive Review on Generative AI for Education," *IEEE Access*, 2024.
- [6] J. Zhang, M. L. Wong, and K. R. Gupta, "Large Language Models for Automated Curriculum Design: A Systematic Review," *Journal of Educational Technology Systems*, vol. 53, no. 1, 2024.
- [7] R. S. Karthik and S. Meena, "Design of an Intelligent Courseware Generator using Retrieval-Augmented Generation (RAG)," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 4, 2024.
- [8] P. Herrera-Sosa et al., "Hyper-Personalization in Higher Education: Leveraging Generative AI for Dynamic Lesson Planning," *Computers and Education: Artificial Intelligence*, vol. 8, 2025.
- [9] T. N. Nguyen and V. H. Pham, "Automated Micro-learning Content Generation via Multi-Agent AI Systems," *IEEE Transactions on Learning Technologies*, vol. 18, no. 2, 2025.
- [10] S. Chatterjee and D. Bhaumik, "Integrating Bloom's Taxonomy into AI-Driven Course Builders for Enhanced Pedagogy," *Asian Journal of Distance Education*, vol. 20, no. 1, 2025.
- [11] M. A. Al-Fahad, "Evaluating the Efficacy of GPT-4 in Generating STEM Course Syllabi and Assessment Rubrics," *Education and Information Technologies*, vol. 30, no. 3, 2025.
- [12] L. Chen, Y. Wang, and H. Kim, "A Framework for Ethical AI in Automated Course Generation and Content Curation," *Ethics and Information Technology*, vol. 28, no. 2, 2025.
- [13] G. D. Roberts et al., "Adaptive Learning Paths: Real-time Curriculum Adjustment using Reinforcement Learning from Human Feedback (RLHF)," *Journal of Artificial Intelligence in Education*, vol. 36, no. 1, 2026.
- [14] K. S. Lakshmi and R. Venkatesh, "Self-Evolving Course Content: The Role of Generative AI in Lifelong Learning Platforms," *International Research Journal of Engineering and Technology (IRJET)*, vol. 13, no. 4, 2026.
- [15] O. J. Adebawale et al., "Multimodal AI Course Generators: Integrating Video, Text, and Interactive Simulations," *Interactive Learning Environments*, 2026..