# A Framework for Applying Generative AI in Educational Course Creation

Sahil Pulikal
Student,
Datta Meghe College of Engineering,
Airoli, Navi Mumbai.

Nita Patil, PhD
Associate Professor,
K. C. College of Engineering,
Thane, Navi Mumbai.

C.M. Raut, PhD
Professor,
Datta Meghe College of Engineering,
Navi Mumbai, India

Sanjay M. Patil, PhD
Professor,
Datta Meghe College of Engineering,
Navi Mumbai, India

## ABSTRACT

Course generator applications have the potential to revolutionize the way that courses are created. However, existing course generator applications suffer from a number of limitations, including lack of customization such as inability to add custom required topics, limited range of subjects, lack of YouTube video integration, and lack of interactive quizzes. This paper presents a new AI course generator framework that addresses the limitations of existing applications.

The framework empowers users to generate highly personalized courses across diverse subject domains. It leverages advanced Generative AI models like those from OpenAI's GPT model and prompt engineering techniques to analyze user input and automatically create comprehensive course outlines, including key topics, and sub topics. Additionally, the framework seamlessly integrates relevant YouTube videos and automatically generates interactive quizzes to enhance engagement and assess student comprehension.

This paper details the proposed framework's architecture and functionalities. The proposed framework holds significant promise for transforming the way educators and learners interact with educational content, leading to a more engaging and personalized learning experience.

## Keywords

Course generator, Generative AI, Artificial Intelligence, Education

## 1. INTRODUCTION

One of the most intriguing and transformative branches of AI is generative AI, which empowers machines to autonomously create content, imitating human-like creativity and problem-solving [1,2]. This paper presents AI powered Course generator architecture representing a significant stride in redefining the way educational contents are created, structured, and enhanced. Proposed here is a comprehensive full stack application framework that leverages Generative AI to dynamically generate course structures, enriches them with relevant YouTube video resources, and augments learning through concept-check quizzes.

Traditionally, crafting educational courses has been a laborious and time-consuming process. Instructors often grapple with the intricate task of designing cohesive course structures that effectively convey knowledge. Moreover, keeping course content updated and engaging poses an ongoing challenge. As the internet becomes a reservoir of educational resources, accessing pertinent external materials, such as YouTube videos, has become indispensable. Lastly, formative assessment in the form of quizzes is crucial for gauging student comprehension and reinforcing learning. This proposed framework seeks to address these multifaceted educational challenges.

This paper's principal goal is to offer a framework for an innovative, user-friendly platform that facilitates the efficient creation of comprehensive courses powered by Generative AI. It can achieve this through the following core functionalities:

1. AI-Driven Course Structure Generation: Utilizing advanced AI models such as Open AI's GPT 3.5 turbo, the framework generates comprehensive course structures by analyzing the input keywords. This not only simplifies course creation but also ensures logical and effective content organization.
2. YouTube Video Integration: The platform dynamically fetches and embeds relevant YouTube videos within the course structure. This enhances learning by providing students with multimedia resources that supplement textual content, catering to diverse learning styles.
3. Concept Check Quiz Generator: To bolster comprehension and engagement, the framework automatically generates concept-check quizzes. These quizzes are aligned with course content, enabling students to reinforce key concepts.

This framework's methodology leverages OpenAI's GPT 3.5 turbo model to analyze user input such as input topics and sub topics to generate comprehensive course outlines. Users provide a course title and subtopics, which are processed through advanced prompts to guide the OpenAI GPT API in generating chapters, subtopics, and quizzes. The framework integrates YouTube video fetching and summary generation based on video transcripts, further enriching the learning experience. User authentication and payment integration ensure secure access and unlimited course generation. A user-friendly interface allows for easy navigation and interaction with course content.

This AI Course Generator framework holds profound significance in the realm of education by offering the following advantages:

1. Efficiency: By automating course creation, educators can significantly reduce the time and effort required for curriculum development, allowing them to focus on instructional quality and innovation.
2. Quality: The integration of external multimedia resources, like YouTube videos, enriches course content, making it more engaging and reflective of real-world scenarios.
3. Assessment: Formative assessment through concept-check quizzes enhances student learning outcomes by providing immediate feedback and opportunities for reinforcement.

This AI Course Generator framework makes several novel contributions to the field of educational technology. Firstly, it utilizes advanced AI models, like OpenAI's GPT, to dynamically generate comprehensive course structures with diverse subject coverage, exceeding the capabilities of existing applications. Secondly, it automatically fetches relevant YouTube videos, enriching the learning experience and catering to diverse learning styles. Most notably, the framework employs a novel approach of generating quizzes directly from YouTube video transcripts, offering an interactive assessment method not found in existing solutions. This integration with OpenAI's GPT API ensures efficient and high-quality content creation, further solidifying the framework's potential to revolutionize education. Additionally, the framework breaks new ground by automatically generating interactive quizzes directly from video transcripts, an innovative approach not found in existing frameworks. This feature promotes deeper engagement and knowledge retention by seamlessly integrating assessment quizzes with learning materials.

This paper is organized to offer a thorough comprehension of the AI Course Generator framework. The related works on Generative AI are presented in Section 2. Some limitations of existing systems are highlighted in Section 3. Fundamentals and Basics of Generative AI are discussed in Section 4. Proposed Methodology is described in Section 5. Results are discussed in Section 6. Conclusion & future scope are discussed in Section 7.

## 2. LITERATURE REVIEW

This section offers a thorough summary of Generative AI and its numerous applications.

R. Gozalo-Brizuela et al. [1] provides study of over 350 applications using generative AI applications showcasing a wide range of unimodal and multimodal Generative AI applications. It offers a structured taxonomy for categorizing them. It also highlights that it is important to consider ethical implications as well as potential pitfalls associated with use of these Generative AI applications.

R. Gozalo-Brizuela et al. [2] provides a taxonomy and assessment of the most widely used generative models published recently. This paper focuses on uses of the models and the kinds of content they produce. It categorizes Gen AI models into nine categories such as texts to images (DALLE-2). It also highlights some limitations for Generative AI models such as it requires a lot of computing resources to run.

P. Sridhar et al. [3] focuses on generating Learning objectives using GPT-4. Results show that the GPT-4 model to a greater extent operates on expected levels of Bloom's Taxonomy. Fair agreement was reached when LOs were classified into each of the six distinct levels of Bloom's Taxonomy by human raters, with an average agreement of 0.31. There was high agreement between the BERT and human classification of LOs, as seen by the agreement of 0.62 between the majority-vote annotation and the BERT classification.

Tsai et al. [4] uses the BERT and GPT-2 pre-trained model for automatically generating short answer questions in Python programming courses in college. Accuracy achieved before fine tuning was 94% and after fine tuning the model was 98%. Notable point was that before fine tuning the model didn't find words related to Python but after fine tuning the model's performance improved and was able to learn words related to Python. Hence, after fine tuning, the effectiveness of the model was improved. This paper provides insights on quiz generation using pretrained models.

C. Zhang et al. [5] provides a comprehensive review of Generative AI. Google Trends report shows the recent increase in the use of Generative AI. It also discusses the underlying techniques used in Generative AI such as Backbone architectures, Self-supervised pre-training techniques categorized according to the training data type.

H. Gimpel et al. [6] gives advice to educators on how to properly utilize ChatGPT and other generative AI tools in the classroom. It provides prompt writing techniques and recommendations for ChatGPT which will help in writing prompts for this framework. It also highlights the risks of using ChatGPT such as generated answers are from non-existent sources. It also showcases the prompts required for quiz generation which will help in this framework.

E. Tajik and F. Tajik [7] offers a sophisticated comprehension of the suitability of GPT in the classroom for educators, instructors, and administrators. It emphasizes how important it is to prepare lesson plans, provide quiz and exam questions, describe class syllabuses, and give instructions for learning activities.

Jiahong Su et al. [8] explains how to apply the IDEE framework to the usage of generative AI in education. It also draws attention to a few difficulties, such as the price of integrating ChatGPT in the classroom and ongoing maintenance and support required. It also highlights some key advantages of using ChatGPT in education such as providing personalized learning experiences for students and supporting teachers in answering student questions.

Aydın et al. [9] explains Generative AI and its fundamentals such as Tokens, Language Models (LMs) & Large Language Models (LLMs). It highlights, with examples, some issues with ChatGPT such as it provides different, incorrect and made-up answers with non-existent references to questions when asked to it repeatedly.

C. Jeong et al. [10] focuses on implementation of Generative AI Services in Enterprise applications. Prompt Engineering allows to generate responses without extensive need of parameters updates via extensive data collection or process fine-tuning. Using prompt engineering techniques, the model's learning and inference abilities are enhanced

C. Jeong [11] investigates the fusion of data and analytics with generative AI. It discusses Enterprise Application integration by introducing middleware tools, Pipeline and ETL and BI reporting.

M. Chui et al [12] provides a comprehensive overview on the economic potential of Generative AI. It shows that there will be a potential economic impact on the global economy of ~15–40% from generative AI use cases.

P. Dhoni [13] explores how Generative AI can be used in IT workplaces. It talks about Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) as examples of Generative AI models.

Qadir, Junaid [14] explores promises and pitfalls of using Generative AI in education. It showcases some applications of ChatGPT such as code generation. It also provides some sample questions and answers asked to ChatGPT. It highlights some shortcomings of ChatGPT such as hallucinative misinformation and lack of reliability

D. Dasgupta et al. [15] walks through the historical perspective of Generative AI. It talks about evolutionary computing, Genetic programming, fundamentals of Generative AI and Generative AI pipeline.

C. K. Y. Chan et al. [16] sheds light on diverse students' perceptions on Generative AI technologies like ChatGPT in higher education by conducting online survey questionnaires consisting of both open ended and closed ended questions.

S. Grassini [17] explores opportunities, challenges and threats posed by ChatGPT in education and possible actions and measures for mitigation in reaction to ChatGPT's implications. Some potential uses of Generative AI include automatic grading system, translating educational materials, individual tutoring, personal mindset coaches, creating quizzes and exam syllabus. Some challenges and threats have been highlighted such as job cuts, accuracy and reliability.

D. Tosic et al. [18] provides detailed technical implementation on implementation of OpenAI API in existing Content Management System. It discusses prompt engineering techniques and agile project methodology. It shows technical implementation, testing, and parsing of OpenAI API.

S. Kim et al. [19] studies on utilizing ChatGPT as a second language learning tool. It showcases some prompts to be used along with their outputs for designing course content, sequences, structures, then designing course units, assessments, etc. It highlights concerns like ChatGPT being incapable of providing detailed feedback and risk of ChatGPT being abused which can possibly hinder learning among students.

**Comparative Analysis of existing Course Generator Frameworks:**

**Table 1: Comparative Analysis of existing Course Generator Frameworks**

| Parameters | Content generation using | YouTube Video integration | Quiz generation based on YouTube transcripts | Customization options |
|---|---|---|---|---|
| EdApp's AI Create | Unknown AI model | No | No | Limited |
| Coursebox | Unknown AI model | No | No | Limited |
| Mini Course Generator | Unknown AI model | No | No | Limited |

## 3. LIMITATIONS OF EXISTING SYSTEM

Some limitations in the existing systems found are listed below:

Lack of customization: Existing course generator applications often lack the ability to generate highly customized courses. This is because they are typically trained on large datasets of existing courses, which may not reflect the specific needs of all users.

Limited range of subjects: Most course generator applications are limited to a small range of subjects, such as math, science, and English. This means that they may not be suitable for users who want to generate courses on more specialized topics.

Lack of YouTube Video Integration: Existing systems lack YouTube video integration which will make learners understand the topic better.

Lack of interactive quizzes: Most course generator applications do not generate interactive or engaging quizzes. This can make it difficult for students to learn and remember the material effectively.

Complicated user interface: makes it difficult for users to generate courses.

To address these limitations, this framework proposes several new features, including:

Customization: Users can customize their courses by providing options to choose the specific topics they want to cover.

Wide range of subjects: Users will be able to generate courses on a wide range of topics.

YouTube Video Integration: YouTube video integration will make learners understand the topic better.

Interactive quizzes: Interactive quizzes in the form of MCQs will help the learner to imbibe the knowledge and help retain the knowledge.

Easy to use interface: Easy to use user interface will help anyone to easily generate quizzes.

## 4. FUNDAMENTALS OF GENERATIVE AI

The area of artificial intelligence known as "Generative AI" is dedicated to creating algorithms that can produce new information, like text, images, audio, and video. Large datasets of pre-existing content are used to train generative AI models, which then learn to produce new content that is comparable to the training set. Many tasks can be accomplished with generative AI models [1], such as:

### 4.1 Text generation models

Text generation models can generate text, such as news articles, creative writing, and code. They are typically based on large language models (LLMs), which are trained on massive datasets of text and code. Text that is comparable to the text that LLMs were trained on can be produced by LLMs, and they can also be used to generate new and creative text formats.

Some examples of text generation models include:

GPT-3: GPT-3 is an LLM developed by OpenAI. It is one of the most powerful text generation models available, and it can produce text in a variety of formats, such as scripts, code, and poetry, musical pieces, email, and letters. [1] [5]

Codex: Codex is an LLM developed by Google AI. It is specifically designed to generate code, and Python, JavaScript, Go, and C++ are just a few of the programming languages in which it may generate code. [1]

LaMDA: LaMDA is an LLM that Google AI has created. It provides users with complete and insightful answers to questions, even if they are unusual, difficult, or open-ended. [1]

### 4.2 Image generation models

Image generation models can generate images, such as realistic photos, paintings, and cartoons. Usually, they are built using deep learning methods like generative adversarial networks (GANs). Two neural networks that have been taught to compete with one another are called GANs. New images are produced by the generator, the first neural network. The discriminator, the second neural network, attempts to discern between authentic images and false ones produced by the generator. The generator gains the ability to produce images that are identical to genuine photos over time.

Some examples of image generation models include:

Imagen: Google AI created the Imagen model, which generates images. It can take text descriptions and turn them into realistic visuals. [1]

DALL-E 2: OpenAI created the DALL-E 2 image generating model. It can alter pre-existing photos and produce realistic visuals from text descriptions. [1]

Parti: Parti is an image editing model developed by Google AI. It can be used to inpaint missing pixels in images and to generate different variations of a given image. [1]

### 4.3 Video generation models

Video generation models can generate videos, such as realistic videos, music videos, and movies. Although they are still in the early phases of development, they have the power to completely alter the way videos are produced.

Some examples of video generation models include:
DALL-E 2: It can generate videos from text descriptions. [1]
Imagen: Imagen can also generate videos from text descriptions, but the videos are not as realistic as those generated by DALL-E 2. [1]

## 4.4 Text-to-3D models

Text-to-3D models can create realistic and creative 3D images by inputting text. Some examples of Text-to-3D models include Dreamfusion and Magic3D [1].

## 4.5 Speech synthesis

Text may be converted into speech that sounds human using GAI speech synthesis models. Usually, these models are built using deep learning methods like recurrent neural networks (RNNs). For example, the Google Cloud Text-to-Speech API can be used to generate human-like speech from text [1].

## 4.6 Overview of generative AI backbones

Generative AI backbones are the underlying architectures that generative AI models are built on. They provide a general framework for training generative models, and they may be utilized to produce a wide range of content kinds, such as text, pictures, and audio.
Each generative AI backbone has its own strengths and weaknesses. While GANs can produce text, audio, and images of excellent quality, they can also be unstable and challenging to train. VAEs can produce less high-quality text, audio, and images than GANs, but they are more reliable and easier to train. While transformers can produce text, graphics, and code of excellent quality, training them computationally can be more costly than training GANs or VAEs.
The choice of generative AI backbone depends on the specific task at hand. GANs might be the greatest option if producing high-quality text, audio, or images is the main objective. Transformers might be the best option if producing high-quality text, images, or code is the aim and computational resources are available. VAEs might be the best option if the objective is to produce high-quality text, images, or audio and computing resources are limited or if model stability is crucial.

## 4.7 Generative Adversarial Networks (GANs)

One of the most widely used generative AI backbones is the GAN. The generator and discriminator neural networks make up a GAN. The discriminator is in charge of differentiating between actual data samples and fictitious data samples produced by the generator, while the generator is in charge of producing new data samples.
Because GANs are taught adversarially, the generator and discriminator are always attempting to outwit one another. The discriminator looks for differences between actual and false data samples, while the generator attempts to create false data samples that are identical to real data samples.
It has been demonstrated that GANs are capable of producing text, audio, and images of excellent quality. They can, however, be unsteady and challenging to train.
Deep fakes and realistic images for video games and movies are produced using GANs to create realistic representations of people, places, and objects.

## 4.8 Variational Autoencoders (VAEs)

Another well-liked generative AI backbone is VAEs. An autoencoder, or neural network taught to recreate its input data, is a kind of VAE. Along with learning to reconstruct the input data, VAEs also pick up a latent representation of the data during training.
A low-dimensional representation of the data that captures its key characteristics is called the latent representation. By taking samples from the data's latent representation, VAEs are able to produce fresh data samples.
Compared to GANs, VAEs are more stable and require less training. They might not, however, be able to produce text, audio, and images with the same level of quality as GANs. VAEs are being used to generate new drug molecules, to create personalized learning experiences for students, and to develop new materials. [1]

## 4.9 Transformers

Transformers are a relatively new generative AI backbone that has shown promising results. The encoder-decoder architecture, a neural network design frequently employed in machine translation, serves as the foundation for transformers. It has been demonstrated that transformers are useful for producing high-quality text, graphics, and code. Nevertheless, compared to GANs or VAEs, they may require more computing resources to train.
For composing news articles and creating code for new software applications, transformers are utilized to produce realistic text and code. [1]

## 4.10 Benefits of Generative AI

There are several potential benefits of generative AI, such as:
Enhanced productivity: Many tasks now performed by humans, like producing marketing materials, writing reports, and developing new goods and services, can be automated by generative AI. This can free up human laborers to concentrate on more strategic and creative work. [1, 5]
Enhanced creativity: By giving humans fresh concepts and motivation, generative AI can enhance creativity. Generative AI may be utilized, for instance, to create original stories, songs, and product designs. [1]
Customized experiences: Users can have individualized experiences thanks to generative AI. Generative AI can be used, for instance, to provide consumers with individualized learning experiences or product and service recommendations. [1]
Improved decision-making: Generative AI can help people to make better decisions by providing them with insights into complex data. Generative AI may be used, for instance, to find patterns in data or forecast future events. [1]
New products and services: It is now possible to develop previously unattainable new products and services. For instance, generative AI may be utilized to produce novel medications and medical treatments as well as innovative teaching resources. [1]

## 4.11 Challenges of Generative AI

Generative AI confronts difficulties, such as:
Requirements for data: To train, generative AI models need a lot of high-quality data. The collection, labeling, and processing of this data can be costly and time-consuming.
Requirements for computation: Training and deploying generative AI models can be computationally costly. This may make it more difficult for small companies and groups to get started.

Biases: As a result of reflecting the biases in the training data, generative AI models may exhibit biases. This may result in the production of objectionable or dangerous content.
Misinformation: Deep fakes and other types of misinformation can be produced using generative AI models. This may be used to harm someone's reputation or to manipulate others.
Safety: Content that is harmful or dangerous, like weapons or synthetic medicines, can be created using generative AI models. It is imperative to guarantee the safe and responsible application of generative AI models.
The creation and application of generative AI are fraught with ethical issues in addition to these difficulties. For instance, it's critical to think about the effects generative AI will have on society and how it will be utilized to produce and disseminate content.

## 4.12 Addressing the challenges of Generative AI

There are a number of ways to address the challenges [5][8] of generative AI. For example:
Data requirements: Researchers are developing new techniques for training generative AI models with less data. Additionally, there is a growing movement to create open-source datasets that can be utilized for generative AI model training.
Computational requirements: Researchers are developing new algorithms and hardware that can make generative AI models more efficient. Additionally, cloud computing providers are now offering services that can make it easier and more affordable to train and deploy generative AI models.
Bias: Researchers are developing new techniques for mitigating bias in generative AI models. Additionally, it is important to carefully curate the data that is used to train generative AI models.
Misinformation: Researchers are developing new techniques for detecting and preventing the creation of deepfakes and other forms of misinformation. Additionally, it is important to educate the public about the potential for deep fakes and other forms of misinformation.
Safety: New methods are being developed by researchers to guarantee the responsible and safe application of generative AI models. Establishing ethical standards for the creation and application of generative AI is also crucial.

# 5. PROPOSED METHODOLOGY

The proposed methodology consists of framework architecture and technical implementation which are explained as follows

## 5.1 Proposed framework architecture

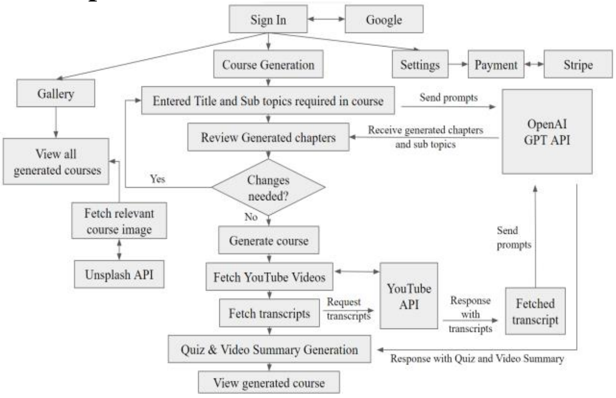


**Fig 1: Proposed framework architecture**

The proposed framework architecture has many components which are shown in Fig. 1 and explained as follows

### *5.1.1 User Authentication:*

User sign-in using Google authentication.
Google authentication credentials are verified and validated.
Upon successful validation, a unique user session is created and associated with the user's account.

### *5.1.2 Course Generation:*

User enters the title and subtopics required for the course.
User input is validated to ensure all required fields are entered correctly.
Advanced prompts are generated based on the user input and sent to the OpenAI GPT API.
OpenAI GPT API processes the prompts and generates corresponding chapters and subtopics.
Generated chapters and subtopics are received from the OpenAI GPT API and displayed to the user for review.
User reviews the generated content and has the option to modify the title and subtopics if necessary.
If modifications are required, the user is directed back to the input stage.
If the user accepts the generated content, the course generation process proceeds.

### *5.1.3 User Interaction with the AI Model and Prompt Engineering:*

Generate advanced prompts based on the user input and send them to the OpenAI GPT API.
Receive generated chapters and subtopics from the OpenAI GPT API and display them to the user for review.
Allow the user to review the generated content and modify the title and subtopics if necessary.
Users can indirectly interact with the AI model by providing input in the course generation page which takes in input from users such as main topics and sub topics of the required course and the system forms the following prompts based on user inputs. Depending on the intended result, the prompts can be straightforward or intricate. For example, system could generate the following prompt:

```
"You are an AI capable of curating course content, coming up with relevant chapter titles, and finding relevant
youtube videos for each chapter",
new Array(units.length).fill(
  `It is your job to create a course about ${title}. The user has requested to create chapters for each of the
  units ${units.join(
    ", "
  )}. Each unit should contain more than one relevant chapers related to that particular unit and ${title}. Then,
  for each chapter, provide a detailed youtube search query that can be used to find an informative educational
  video for each chapter. Each query should give an educational informative course in youtube.`
```

**Fig. 2: Prompt for course creation**

The above prompt uses prompt engineering techniques such as Role based prompting giving system input instructing and assigning a unique persona to AI model and Task-based instructions which considers user's input containing required course and requesting to generate course content.
The AI model would then generate a course structure with chapters and subtopics based on the input prompt. The user could then review the generated content and provide feedback. The feedback could be used to improve the generated content or to generate new content altogether.

### *5.1.4 User Feedback:*

User feedback is incorporated into the framework in two ways:
If the user accepts the generated content, the course generation process proceeds.
If modifications are required, the user is directed back to the input stage.

#### *5.1.5 YouTube Video and Transcript Fetching:*

Recursively fetch related videos for each chapter and subtopic, fetch the relevant YouTube video URL using the YouTube Data API.

Asynchronously download the video transcripts for each fetched YouTube video.

#### *5.1.6 Quiz and Video Summary Generation:*

Generate advanced prompts based on the fetched video transcripts.

Send the advanced prompts to the OpenAI GPT API to generate quizzes and YouTube video summaries.

Receive generated quizzes and video summaries from the OpenAI GPT API and associate them with the corresponding generated chapters and sub topics.

#### *5.1.7 Course Visualization:*

Present the generated course to the user, including chapter and subtopic details, relevant YouTube videos, quizzes, and video summaries.

Provide a user-friendly interface for navigating the course content and interacting with quizzes and video summaries.

#### *5.1.8 Payment Integration:*

Implement Stripe payment processing to enable users to generate an unlimited number of courses.

Integrate Stripe payment gateway into the application to handle secure online transactions.

Upon successful payment, grant access to the unlimited course generation.

#### *5.1.9 Course Gallery:*

Display a gallery of all generated courses accessible to the user.

Fetch relevant thumbnail images for each generated course based on course title using the Unsplash API.

Display the course thumbnail image, title, and sub topics for each generated course in the gallery.

The proposed framework is different from existing methods in the following ways:

1. It uses the latest AI models, such as OpenAI GPT-3 model, which is a state-of-the-art AI model for generating text and to generate high-quality course structures.
2. It allows users to provide feedback on the generated content, which can be used to improve the output.
3. It integrates with YouTube to provide users with access to relevant videos for a more engaging and interactive learning experience.
4. It allows users to create interactive quizzes based on video content transcripts, which can help learners assess their understanding of the material and reinforce key concepts.
5. It is scalable and can be used to generate courses on a wide range of topics.
6. It is made to be user-friendly and accessible to users with all levels of technical expertise.

The proposed framework has the following limitations and boundary conditions:

1. User input and system-generated prompts have an impact on the quality of the content that is generated.
2. The AI model may not be able to generate content on all topics.
3. The AI model may generate content that is biased or inaccurate.
4. The framework requires a stable internet connection.

## 5.2 Proposed technical implementation

### *5.2.1. Setting up a new folder for NextJS and configuring it.*

This involves creating a new folder for NextJS and installing the necessary dependencies [30].

Once the folder has been created, one needs to install the necessary dependencies.

#### *5.2.1.1. Configure NextJS.*

One can do this by creating a next.config.js file. This file will contain the NextJS configuration, such as the routing configuration and the TailwindCSS configuration.

#### *5.2.1.2. Create a database.*

Database is needed to store the course data and the user data. Any database can be used, such as MySQL, PostgreSQL, or MongoDB. PlanetScale will be recommended to create a free cloud hosted MySQL Database.

#### *5.2.1.3. Create a User Interface.*

Create a UI to allow users to interact with the application framework. The UI will allow users to create courses, view generated courses, and upgrade to paid user by making payment.

#### *5.2.1.4. Use TailwindCSS for styling.*

This involves installing TailwindCSS and using it to style the application.

To use Tailwind CSS, Node.js SDK need to be installed.

Once the SDK is installed, create a tailwind.config.js file in the root folder. This file will contain the TailwindCSS configuration.

Once a configuration file is created, TailwindCSS can be used for styling.

### *5.2.2. Integrating OpenAI model to generate course content.*

This involves creating an OpenAI account and obtaining an API key. The API key will then be used to integrate with OpenAI and generate course content.

Make an OpenAI account and get an API key in order to integrate the OpenAI model. Go to the OpenAI website [https://platform.openai.com/signup] and sign up for a free account.

Once the API key is received, install the OpenAI Node.js SDK. Once the SDK is installed, create a new OpenAI client.

Then use the client to generate course content.

Here prompt engineering techniques can be used for feeding the model with prompts for modifying the output as per requirements.

The outline can then be used to create the course content

### *5.2.3. Integrating YouTube videos:*

To enhance the course content and provide a more comprehensive learning experience, consider integrating the YouTube API into the framework.

Obtain a YouTube API key: To access the YouTube API, create a Google Cloud Platform account and enable the YouTube Data API v3. It will provide an API key to make authorized requests to the API. Install the YouTube Data API Node.js SDK and create a new YouTube client.

Search for relevant videos: Integrate the YouTube API into course generation process to search for relevant videos based on the course topic. Use the API to fetch video titles, descriptions, and thumbnails to display within the course material.

Embed videos in the course material: Enhance the course content by embedding relevant YouTube videos directly into the course material. This will provide learners with a more engaging and interactive learning experience.

Create interactive quizzes based on video content: By Leveraging the YouTube API to create interactive quizzes Multiple choice-based questions on the transcript of content of embedded videos. This will help learners assess their understanding of the material and reinforce key concepts.

### *5.2.4. Implement Stripe payments for paid user functionality.*

This involves creating a Stripe account and obtaining an API key. The API key will then be used to implement Stripe payments in the application framework.

To implement Stripe payments, create a Stripe account and obtain an API key. Visiting the Stripe website and signing up for a free account.

Once the API key is received, install the Stripe Node.js SDK. Once the SDK is installed, create a new Stripe client.

### *5.2.5. Deploy the application framework to production in the cloud.*

This involves building the application framework and deploying it to a production Virtual Private server (VPS).

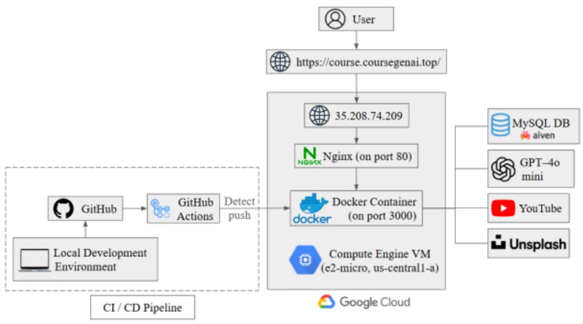


**Fig. 3. Application deployment stack.**

To deploy the application framework to production, one can use a variety of different hosting providers. One may use Google Cloud Platform to deploy this application framework. Docker may be used to containerize application. Nginx may be used for reverse proxy and GitHub Actions for automated CI/CD pipeline.

## 5. RESULTS AND DISCUSSION

Finetuning results are discussed below for gpt model.

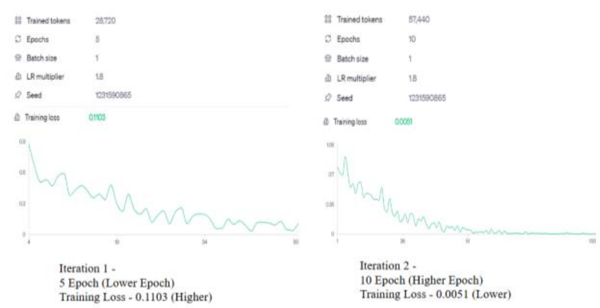


**Fig. 4: Finetuning gpt model with 10 examples**

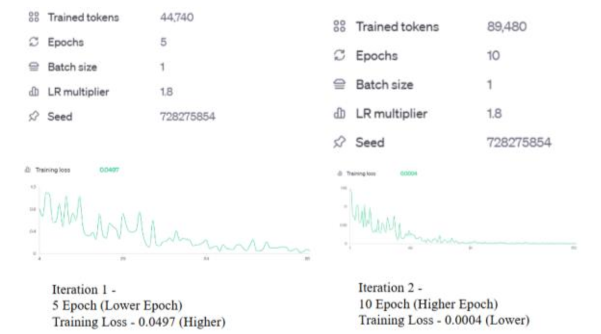


**Fig. 5: Finetuning gpt model with 17 examples**

In first and second scenario, gpt model was finetuned with json lines file containing self-developed dataset of 10 and 17 examples respectively of question and answers related to courses from Bachelors and Masters of Engineering Computer Engineering syllabus from Mumbai University.

**Table 2: Finetuning results.**

| No. of examples | 10 | | 17 | |
|---|---|---|---|---|
| Iteration | 1 | 2 | 1 | 2 |
| Epoch | 5 | 10 | 5 | 10 |
| Training Loss | 0.1103 | 0.0051 | 0.0497 | 0.0004 |

In both scenarios, the quantity of trained tokens roughly doubles when epochs are increased from 5 to 10. There is total more tokens trained with 17 vs 10 examples.

Epochs: Results for 5 epochs (Iteration 1) and 10 epochs (Iteration 2) are displayed in both above figures.

Loss curves: Early on, there are more noticeable variations, but overall, both results display a declining trend in loss over iterations.

Overall, the model performs better when there are 17 samples instead of 10, which reduces the training loss. In both scenarios, training loss is greatly decreased by doubling the number of epochs from five to ten. The optimal results, with the lowest training loss of 0.0004, are obtained by combining more examples (17) and more epochs (10). Given that the relative improvement from 5 to 10 epochs is greater than the improvement from 10 to 17 samples, there seems to be a diminishing returns impact.

These findings imply that increasing the number of epochs and examples can both help with fine-tuning; however, combining more examples with more training epochs yields the best results.

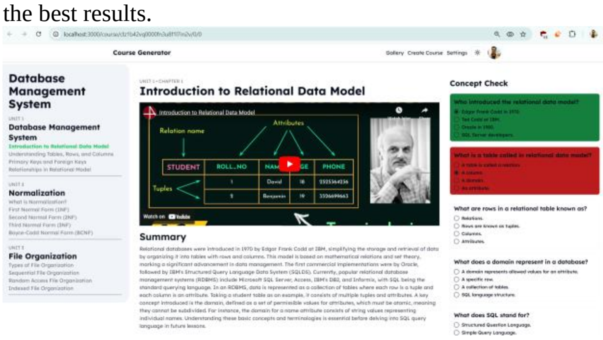


**Fig. 6: Final application output**

Above figure depicts the final generated course for Database Management system topic and it includes a summary, quizzes and YouTube video on the mentioned topic.

## 7. CONCLUSION AND FUTURE SCOPE

This AI powered course generator framework addresses the limitations of existing applications by providing users with the ability to generate highly customized courses on a wide range of subjects, with YouTube video integration and interactive quizzes.

This AI-powered course generator framework automatically generates outlines, fetches relevant YouTube materials, creates quizzes, and summarizes videos, saving time and effort. It ensures the generated outline is tailored to the specific input topics and offers unlimited course generations through Stripe payment integration. The user-friendly course gallery allows browsing generated courses, while the Unsplash API provides visually appealing thumbnails. Advanced prompts and OpenAI GPT API generate comprehensive and engaging content. The framework identifies key concepts, provides summaries, and designs personalized quizzes, enhancing the learning experience. Overall, this framework holds promise for transforming the education landscape.

In future, several exciting avenues exist for further development of this course generator framework:

1. Feature Expansion:

Collaborative Course Creation: Develop functionalities for users to collaborate on creating and sharing courses, fostering a more interactive and engaging learning environment.

Import/Export Functionality: Implement features to import and export courses in various formats, enabling greater flexibility and integration with existing tools.

2. Integration with Educational Platforms:

Learning Management Systems (LMS): Integrate the framework with LMS platforms to facilitate seamless import and export of generated courses, making them readily available within existing educational ecosystems.

Other Educational Tools: Explore integrations with online assessment tools, video conferencing platforms, and other educational resources to create a truly comprehensive learning environment.

3. Future AI Advancements - Multimodal Learning: Explore incorporating other modalities, such as audio and images, into the generated content to create an even richer and more immersive learning experience.

4. Accessibility - Multilingual Support: Implement multilingual functionality to make the framework accessible to a wider audience and break down language barriers to education.

This course generator framework has the power to completely change how courses are made and accessed, democratizing education and opening doors to new and exciting ways of learning.

# 8. REFERENCES

[1] R. Gozalo-Brizuela and E. C. Garrido-Merchán, (2023) "A survey of Generative AI applications," arXiv.org, https://doi.org/10.48550/arXiv.2306.02781.

[2] R. Gozalo-Brizuela and E. C. Garrido-Merchan, (2023), "CHATGPT is not all you need. A state of the art review of large generative AI models," arXiv.org, https://doi.org/10.48550/arXiv.2301.04655.

[3] P. Sridhar et al., (2023), "Harnessing llms in curricular design: Using GPT-4 to support authoring of learning objectives," arXiv.org, https://doi.org/10.48550/arXiv.2306.17459.

[4] Tsai, Danny CL, Willy JW Chang, and Stephen JH Yang. "Short Answer Questions Generation by Fine-Tuning BERT and GPT-2." In Proceedings of the 29th International Conference on Computers in Education Conference, ICCE. 2021.

[5] C. Zhang et al., (2023), "A complete survey on generative AI (AIGC): Is chatgpt from GPT-4 to GPT-5 all you need?," arXiv.org, https://doi.org/10.48550/arXiv.2303.11717.

[6] H. Gimpel et al., (2023), "Unlocking the power of Generative AI models and systems such as GPT-4 and CHATGPT for Higher Education: A guide for students and lecturers," EconStor, https://www.econstor.eu/handle/10419/270970.

[7] E. Tajik and F. Tajik, (2023), "A comprehensive Examination of the potential application of Chat GPT in Higher Education Institutions". TechRxiv, 12-Apr-2023, doi: 10.36227/techrxiv.22589497.v1.

[8] Su, J., & Yang , W. (2023). "Unlocking the Power of ChatGPT: A Framework for Applying Generative AI in Education. ECNU Review of Education", 6(3), 355-366. https://doi.org/10.1177/20965311231168423

[9] Aydın, Ömer and Karaarslan, Enis, (2023), "Is ChatGPT Leading Generative AI? What is Beyond Expectations?" (January 29, 2023). Academic Platform Journal of Engineering and Smart Systems. 11(3), 118-134. DOI: 10.21541/apjess.1293702, Available at SSRN: https://ssrn.com/abstract=4341500 or http://dx.doi.org/10.2139/ssrn.4341500

[10] C. Jeong, (2023), "A study on the implementation of Generative AI services using an enterprise data-based LLM Application Architecture," arXiv.org, https://arxiv.org/abs/2309.01105.

[11] P. Dhoni, (2023), "Exploring the Synergy between Generative AI, Data and Analytics in the Modern Age". TechRxiv, 29-Aug-2023, doi: 10.36227/techrxiv.24045792.v1.

[12] M. Chui et al., (2023), "The economic potential of generative AI The next productivity frontier," 2023. Available: http://dln.jaipuria.ac.in:8080/jspui/bitstream/123456789/14313/1/The-economic-potential-of-generative-ai-the-next-productivity-frontier.pdf

[13] P. Dhoni, (2023), "Unleashing the Potential: Overcoming Hurdles and Embracing Generative AI in IT Workplaces: Advantages, Guidelines, and Policies". TechRxiv, 20-Jul-2023, doi: 10.36227/techrxiv.23696709.v1.

[14] Qadir, Junaid (2022). Engineering Education in the Era of ChatGPT: Promise and Pitfalls of Generative AI for Education. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.21789434.v1

[15] D. Dasgupta, . Deepak . Venugopal and K. D. Gupta, (2023), "A Review of Generative AI from Historical Perspectives". TechRxiv, 17-Feb-2023, doi: 10.36227/techrxiv.22097942.v1.

[16] C. K. Y. Chan and W. Hu, (2023), "Students' voices on Generative AI: Perceptions, benefits, and challenges in Higher Education," arXiv.org, https://arxiv.org/abs/2305.00290.

[17] S. Grassini, (2023), "Shaping the Future of Education: Exploring the Potential and Consequences of AI and ChatGPT in Educational Settings," Education Sciences, vol. 13, no. 7, p. 692, Jul. 2023, doi: 10.3390/educsci13070692.

[18] D. Tosic, (2023), 'Artificial Intelligence-driven web development and agile project management using OpenAI API and GPT technology : A detailed report on technical integration and implementation of GPT models in CMS with API and agile web development for quality user-centered AI chat service experience', Dissertation, 2023. Available from: https://urn.kb.se/resolve?urn=urn:nbn:se:miun:diva-48446.

[19] S. Kim, J. Shim, and J. Shim, (2023), "A study on the utilization of OpenAI CHATGPT as a Second language learning tool," Journal of Multimedia Information System, https://doi.org/10.33851/JMIS.2023.10.1.79.