# GEN AI PROJECT PHASE 1

## Phase 1: Proposal & Idea Submission

### 1. Project Title:

Sentence generator using Generative AI

### 2. Domain:

Generative AI | NLP | Sentence Generation

### 3. Problem Statement:

In Natural Language Processing, generating contextually accurate and grammatically correct sentence continuations is essential for applications like chatbots, writing assistants, and auto-completion tools. This project addresses the challenge by implementing a text generation system that extends user-provided prompts with coherent continuations using transformer-based generative language models..

### 4. Proposed Solution:

This project will implement a **Sentence Generator** using transformer-based pre-trained language models (e.g., GPT-2, GPT-Neo) via the Hugging Face transformers library. The system will:

* Accept a user-provided input: a sentence or short paragraph.
* Use a generative language model to generate one or more additional sentences that continue the input naturally.
* Maintain coherence, relevance, and grammatical correctness in the continuation.
* Optionally allow the user to configure the number of sentences or word count to generate.
* Provide a flexible foundation for applications such as story generation, creative writing, and intelligent auto-completion tools.

### 5. Objectives:

* Build a working prototype that continues user input using GPT-2.
* Demonstrate the effect of different decoding parameters (top\_k, top\_p, temperature).
* Allow users to interactively input prompts and see coherent generated output.

**6. Expected Outcome:**

* A command-line or notebook-based NLP application capable of generating coherent and contextually relevant continuations for a user-provided complete sentence or paragraph.
* A console demo where users input a valid sentence, and the GPT-2 model produces multiple plausible next sentence completions.
* Optional: Implementation of an evaluation component to assess the grammaticality and semantic relevance of the generated continuations.
* The system can serve as a foundational tool for applications such as AI writing aids, automated story generation, and intelligent content suggestion engines.

### 7. Tools & Technologies to be Used:

* Python – Primary programming language
* Transformers library (by Hugging Face) – For accessing pre-trained language models
* Pre-trained models:
* GPT-2 or GPT-2-large for generative text completion
* Optionally GPT-Neo for open-source alternative
* Jupyter Notebook – For model experimentation and testing
* Google Colab / Local GPU setup – For running inference efficiently

### 8. References:

- HuggingFace Transformers Documentation  
- Google BERT Research Paper  
- OpenAI GPT Models Documentation  
- NLP Projects on Next Sentence Prediction on GitHub