Entertainment Content Generator

Course Name: Generative AI

Institution Name: Medicaps University – Datagami Skill Based Course

Student Name(s) & Enrolment Number(s):

Sr no	Student Name	Enrolment Number
1	Aman Sharma	EN22CS301109
2	Ananya Bhatia	EN22CS301117
3	Ankit Nagar	EN22CS301134
4	Anusha Singh Panwar	EN22CS301179
5	Arpit Patidar	EN22CS301199

Group Name: Group 02D2

Project Number: GAI-14

Industry Mentor Name: Mr. Suraj Nayak

University Mentor Name: Dr. Hemlata Patel

Academic Year: Jan – Jun 2026

Problem Statement & Objectives

1.Problem Statement:

The process of developing professional entertainment content such as screenplays, structured narratives, and cinematic story arcs is complex and multi-layered. It typically involves multiple stages including concept ideation, logline formation, pitch drafting, act structuring, character development, and scene writing. Each stage depends heavily on contextual continuity and narrative consistency.

Most existing AI-based content generation tools operate on single-prompt generation mechanisms. They generate isolated outputs without maintaining structured progression or memory of previously generated content. As a result, the generated narratives often suffer from:

- Lack of continuity between story stages
- Inconsistent character behavior and tone
- Disconnected plot progression
- Absence of professional screenplay formatting
- No contextual retrieval mechanism
- Poor structural alignment with industry storytelling standards

Additionally, traditional content creation requires significant time, creative effort, and domain expertise. Emerging creators, writers, and content developers often struggle to transform raw ideas into structured, industry-standard scripts efficiently.

Therefore, there is a need for a system that:

- Breaks content creation into structured narrative stages
- Maintains semantic memory across stages
- Retrieves relevant context intelligently
- Ensures formatting consistency
- Produces professional, structured screenplay outputs

The problem addressed in this project is:

To design and implement a multi-stage, memory-augmented AI-based entertainment content generation system that maintains contextual continuity, enforces screenplay

structure, and transforms raw user ideas into professionally structured narrative outputs.

2.Project Objectives:

The primary objectives of project are:

1. To build a multi-stage screenplay generation pipeline.
2. To implement context continuity using a vector database.
3. To enforce professional screenplay formatting standards.
4. To structure story development in industry-standard order:
    - Concept → Logline → Pitch → Outline → Characters → Scene
5. To integrate large language model (LLM) generation using Gemini.
6. To design a clean, professional Streamlit interface.
7. To implement retry logic for production-grade reliability.
8. To demonstrate integration of:
    - NLP
    - Vector embeddings
    - FAISS similarity search
    - Prompt engineering
    - Context-aware generation

3.Scope of the Project

i. Development of a **multi-stage AI-based screenplay generation system** that converts a user's idea into structured narrative outputs.

ii. Implementation of sequential content generation stages including:

- Concept
- Logline
- Pitch
- Story outline

- Character profiles
- Scene screenplay

iii. Integration of **genre and tone customization** to generate tailored entertainment content.

iv. Use of **vector-based memory (FAISS + embeddings)** to maintain narrative continuity across stages.

v. Enforcement of **industry-standard screenplay formatting rules**.

vi. Development of an interactive **Streamlit-based user interface** for smooth content generation.

vii. Implementation of **LLM integration (Gemini)** with structured prompt engineering.

viii. The project focuses only on **text-based content generation** .

Proposed Solution

1.**Key features**

i. Multi-Stage Screenplay Generation Pipeline

ii. Concept to Scene Structured Workflow

iii. Genre-Based Customization

iv. Tone-Based Content Variation

v. Memory-Augmented Context Retrieval

vi. FAISS-Based Semantic Vector Storage

vii. Context Injection Across Narrative Stages

viii. Industry-Standard Screenplay Formatting Enforcement

ix. Stage-Level Regeneration Capability

x. Google Gemini LLM Integration

xi. Prompt Template-Based Structured Generation

xii. Session-Based Memory Management

xiii. Retry Logic for API Stability

xiv. Interactive Streamlit User Interface

xv. Modular and Scalable System Architecture

2.Overall Architecture / Workflow

Overall Workflow

1. **User Input Submission**
   The user enters a seed idea and selects genre and tone through the Streamlit interface.
2. **Input Validation & Processing**
   The system validates the input, normalizes text, and prepares structured parameters for processing.
3. **Stage Selection**
   The user chooses either:
   - Full screenplay generation pipeline, or
   - A specific narrative stage (e.g., Concept, Logline, Scene).
4. **Context Retrieval from Memory**
   Previously generated outputs are converted into embeddings and stored in FAISS.
   When a new stage runs, relevant past content is retrieved using semantic similarity search.
5. **Prompt Construction**
   The system builds a structured prompt using:
   - Selected genre
   - Selected tone
   - Retrieved contextual memory
   - Stage-specific template
6. **Content Generation (LLM Processing)**
   The constructed prompt is sent to the Google Gemini model to generate structured narrative output.

7. **Output Formatting**
   The generated content is formatted according to industry-standard screenplay rules (scene headings, dialogue structure, etc.).

8. **Memory Storage**
   The newly generated output is embedded and stored in the vector database for future contextual use.

9. **Display to User**
   The final structured output is displayed in the interface, with options for regeneration or continuation.

3. Tools & Technologies Used

i. **Python**
Used as the core programming language for implementing backend logic, pipeline orchestration, and system integration.

ii. **Streamlit**
Used to develop the interactive web-based user interface for input handling, output display, and session management.

iii. **Google Gemini (gemini-2.5-flash)**
Used as the Large Language Model (LLM) for generating structured narrative and screenplay content.

iv. **SentenceTransformers**
Used to generate semantic embeddings of generated text for contextual memory storage and retrieval.

v. **FAISS (Facebook AI Similarity Search)**
Used as a vector database for storing embeddings and performing similarity-based context retrieval.

vi. **Prompt Engineering Framework**
Structured prompt templates designed for each narrative stage to ensure consistency, formatting, and genre alignment.

vii. **Streamlit Session State**
Used for maintaining stage continuity and user interaction within a session.

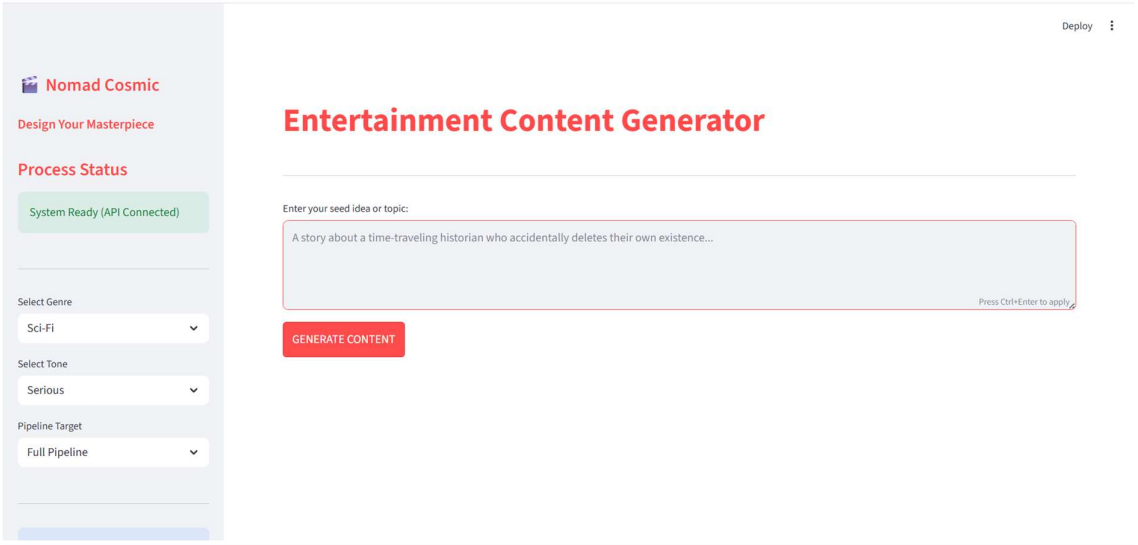
viii. **Dotenv (.env Configuration)**
Used for securely managing API keys and environment variables.

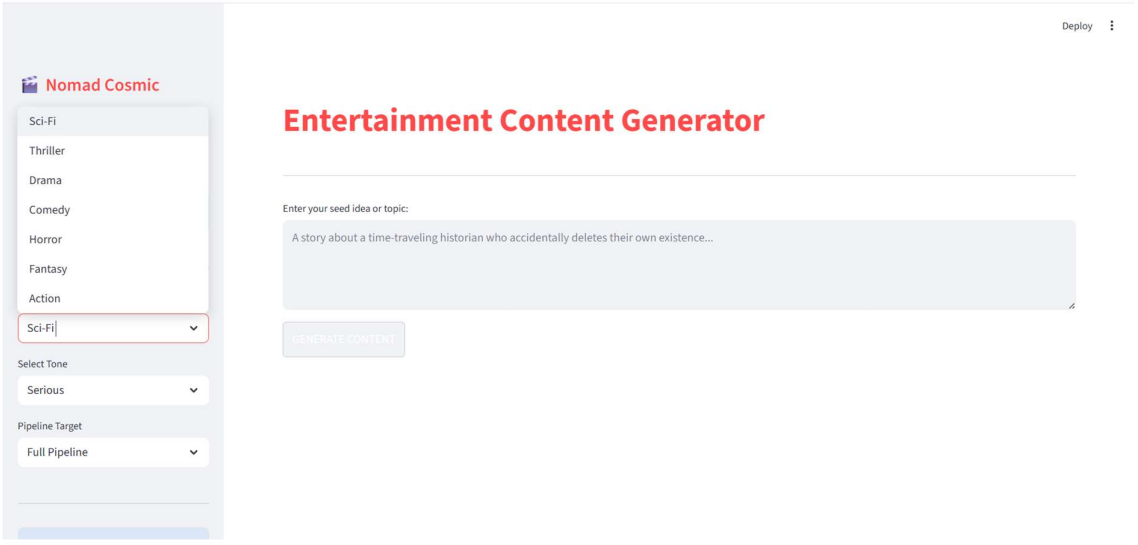
ix. **Retry & Error Handling Mechanism**
Implemented to manage API rate limits and ensure system reliability.

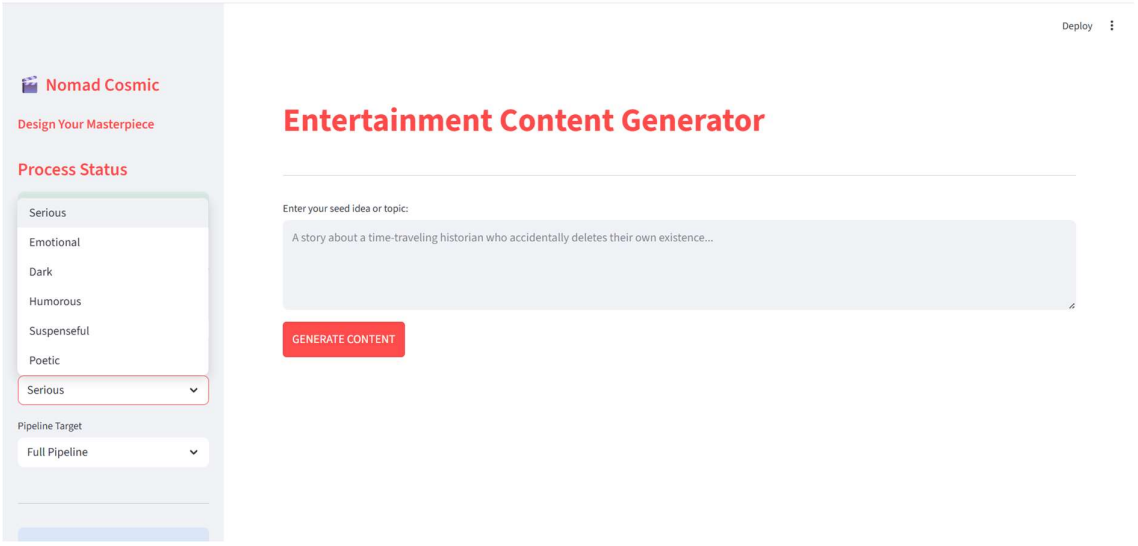
x. **Git (Version Control)**
Used for tracking project development and managing code versions.

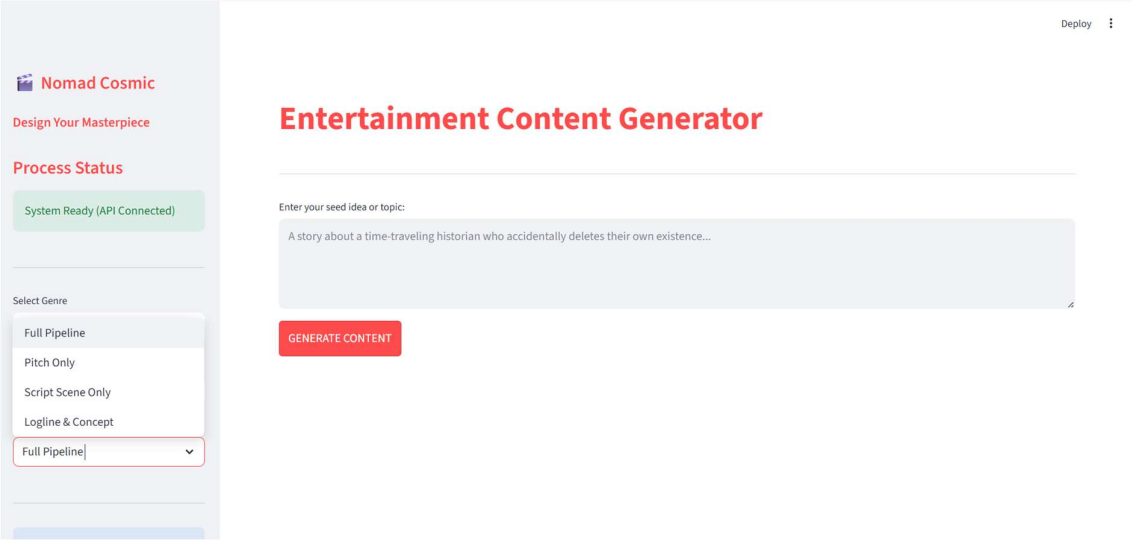
Results & Output

1. **Screenshots / outputs**

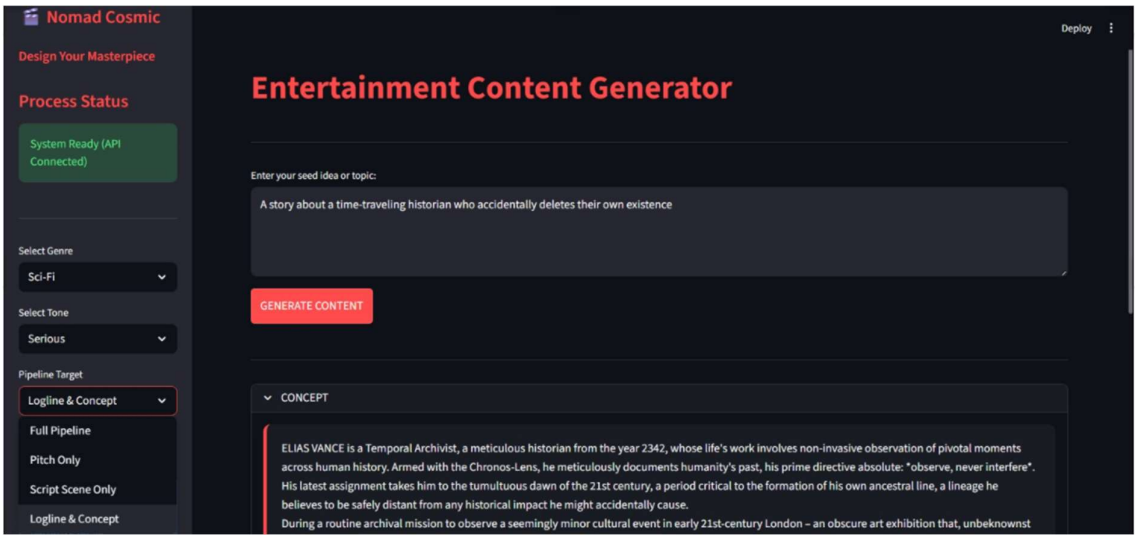

Deploy
Nomad Cosmic
Design Your Masterpiece
Process Status
System Ready (API Connected)
Select Genre
Sci-Fi
Select Tone
Serious
Pipeline Target
Full Pipeline
Entertainment Content Generator
Enter your seed idea or topic:
A story about a time-traveling historian who accidentally deletes their own existence...
Press Ctrl+Enter to apply
GENERATE CONTENT

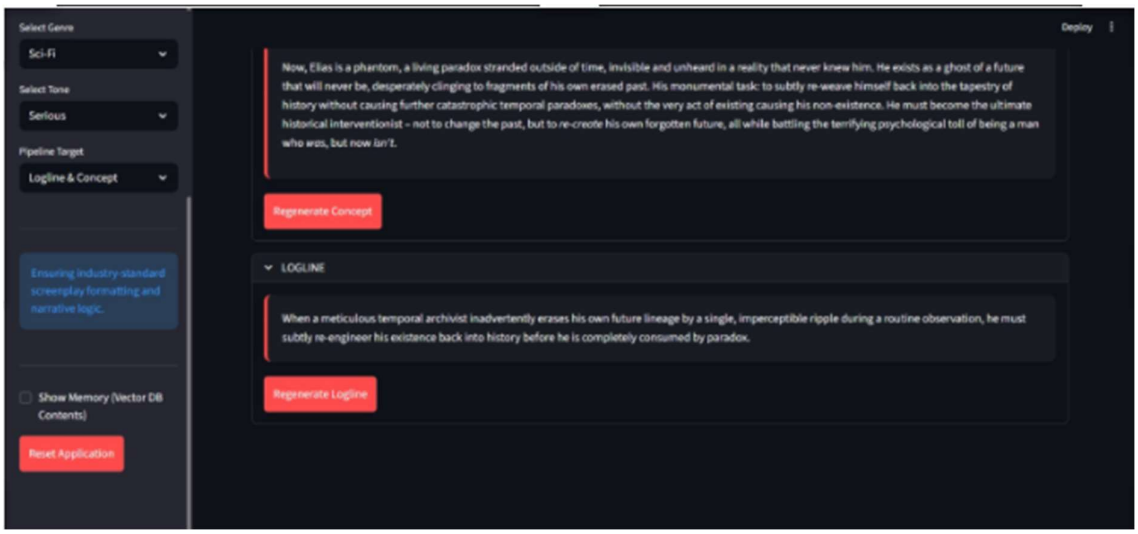

Deploy
Nomad Cosmic
Sci-Fi
Thriller
Drama
Comedy
Horror
Fantasy
Action
Sci-Fi
Select Tone
Serious
Pipeline Target
Full Pipeline
Entertainment Content Generator
Enter your seed idea or topic:
A story about a time-traveling historian who accidentally deletes their own existence...
GENERATE CONTENT


Deploy
Nomad Cosmic
Design Your Masterpiece
Process Status
Serious
Emotional
Dark
Humorous
Suspenseful
Poetic
Serious
Pipeline Target
Full Pipeline
Entertainment Content Generator
Enter your seed idea or topic:
A story about a time-traveling historian who accidentally deletes their own existence...
GENERATE CONTENT



2. Reports / dashboards / models





3. Key outcomes

i. Successfully developed a multi-stage AI-powered screenplay generation system.

ii. Implemented a memory-augmented architecture using vector embeddings and FAISS for contextual continuity.

iii. Achieved consistent narrative flow across multiple storytelling stages (Concept to Scene).

iv. Enforced industry-standard screenplay formatting in generated outputs.

v. Integrated Google Gemini LLM for structured and high-quality content generation.

vi. Designed a modular and scalable pipeline-based system architecture.

vii. Built an interactive Streamlit-based user interface for smooth user interaction.

viii. Implemented semantic similarity retrieval for context-aware generation.

ix. Applied prompt engineering techniques to control tone, genre, and structure.

x. Demonstrated practical implementation of AI, NLP, vector databases, and system orchestration in the entertainment domain.

Conclusion

Nomad Cosmic successfully demonstrates the design and implementation of a multi-stage, memory-augmented AI-based screenplay generation system. The project transforms a simple user-provided idea into a professionally structured narrative by following a sequential storytelling pipeline. By integrating Google Gemini for content generation, Sentence Transformers for semantic embeddings, and FAISS for contextual memory retrieval, the system ensures narrative continuity, genre consistency, and industry-standard screenplay formatting.

Through this project, key learnings include practical understanding of prompt engineering, vector-based similarity search, memory-augmented generation architecture, API integration, session management, and modular system design. The project reflects the effective application of Gen AI concepts in solving real-world creative automation challenges within the entertainment domain.

Future Scope & Enhancements

1. Integration of persistent vector database for long-term memory storage.

2. Implementation of multi-user authentication system.
3. Personalized script libraries for individual users.
4. Real-time collaborative screenplay writing feature.
5. Export functionality in PDF and industry-standard screenplay formats.
6. Cloud deployment for scalability and performance optimization.
7. Advanced analytics dashboard for story and character analysis.
8. User preference learning and adaptive content personalization.
9. Integration of multimedia support (storyboarding / image generation).
10. Fine-tuning domain-specific models for improved screenplay authenticity.
11. Multi-language screenplay generation support.
12. API integration for publishing or external tool connectivity.