Project Report: EduTutor AI

1. INTRODUCTION

1.1 Project Overview

EduTutor AI is an intelligent question-answering application powered by IBM Watsonx's foundation models. It allows users—particularly students—to input academic or general queries and receive AI-generated responses in natural language. The app is built using Streamlit and deployed via Streamlit Cloud, integrating IBM Watson's Granite 3B-Instruct model.

1.2 Purpose

The purpose of this project is to create a virtual AI tutor that can enhance students' learning experiences by providing instant answers, explanations, and insights, especially in self-study environments.

2. IDEATION PHASE

2.1 Problem Statement

Students often struggle to find immediate answers to academic questions outside classroom hours. Most existing platforms are either too generic or require paid subscriptions.

2.2 Empathy Map Canvas

- Think & Feel: Needs accurate, fast answers.
- See: Many irrelevant or overly technical answers online.
- Say & Do: Prefers conversational, to-the-point help.
- Hear: From peers—"I wish I had a 24/7 study buddy."

2.3 Brainstorming

- Use IBM Watsonx models for language understanding.
- Build a light, deployable UI with Streamlit.
- Host on Streamlit Cloud for free/public access.

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. User lands on the app

2. Enters a question

3. Clicks “Get Answer”

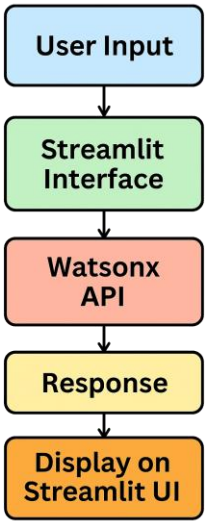
4. Receives an AI-generated response

5. Uses the answer for learning

3.2 Solution Requirement

- API Key and Project ID from IBM Cloud
- Python, Streamlit
- Access to Watsonx models

3.3 Data Flow Diagram



3.4 Technology Stack

- Frontend: Streamlit
- Backend: Python
- AI Model: IBM Watsonx (Granite 3B-Instruct)
- Hosting: Streamlit Cloud

4. PROJECT DESIGN

4.1 Problem Solution Fit

EduTutor AI aligns with the growing need for AI-based education tools that work as personalized tutors.

4.2 Proposed Solution

An intuitive app where users ask questions and get real-time AI responses.

4.3 Solution Architecture

- Streamlit captures user input
- Backend initializes Watsonx model via API
- Generates text response using IBM's foundation model
- Displays it back to the user

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Phase	Duration	Tasks
Week 1	3 days	Setup IBM Watsonx & Streamlit
Week 2	2 days	Model Integration
Week 3	3 days	Streamlit UI + Deployment
Week 4	2 days	Testing + Documentation

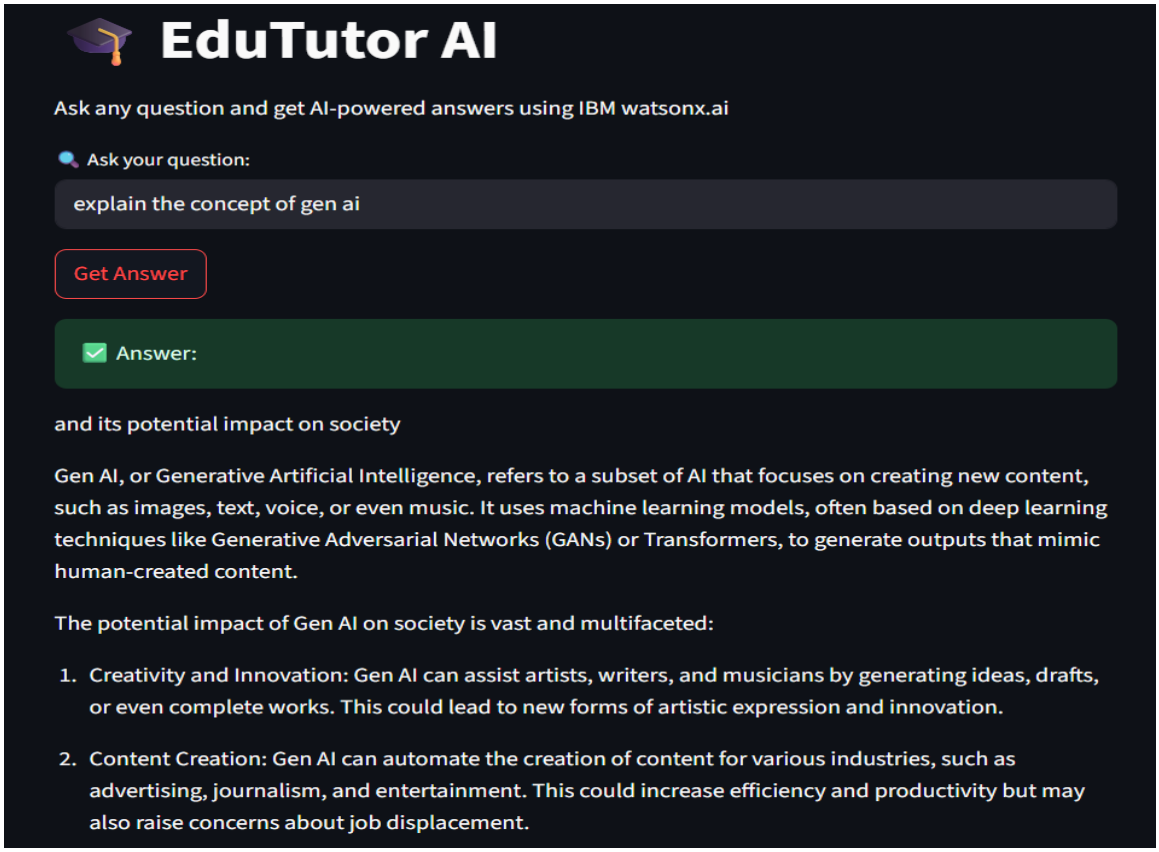
6. FUNCTIONAL AND PERFORMANCE TESTING

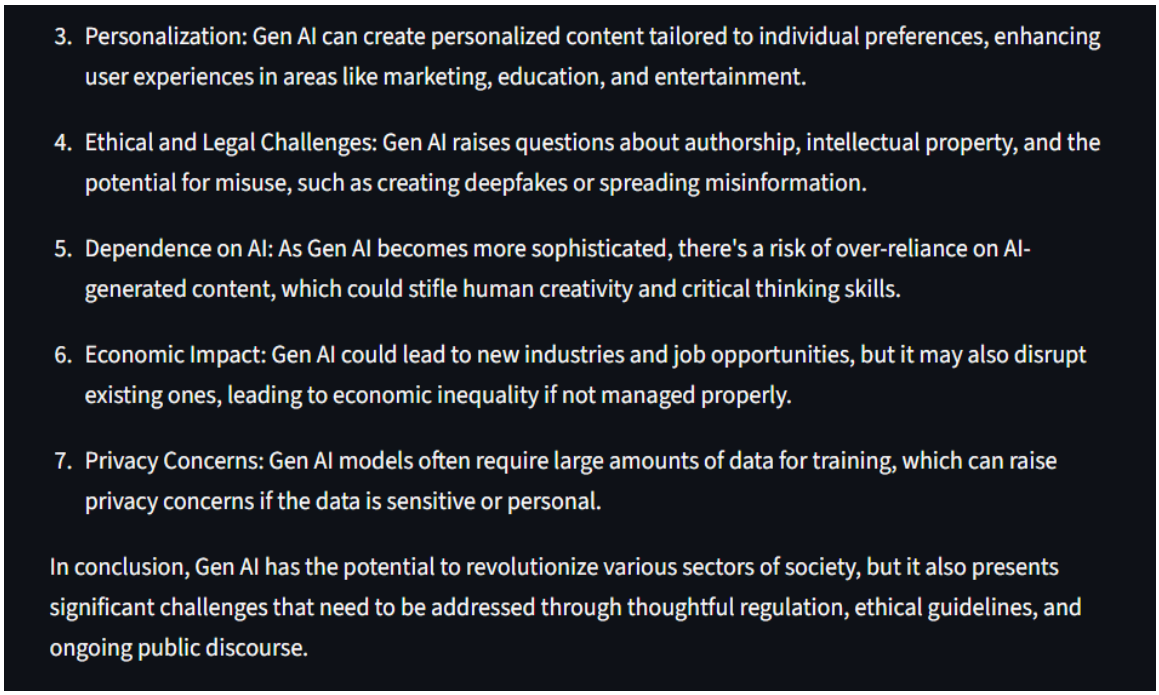
6.1 Performance Testing

Tested the app for:

- Model latency (response within 2-3 seconds on average)
- UI responsiveness on mobile and desktop
- API key validation and error handling

7. RESULTS


EduTutor AI
Ask any question and get AI-powered answers using IBM watsonx.ai
Ask your question:
explain the concept of gen ai
Get Answer
Answer:
and its potential impact on society
Gen AI, or Generative Artificial Intelligence, refers to a subset of AI that focuses on creating new content, such as images, text, voice, or even music. It uses machine learning models, often based on deep learning techniques like Generative Adversarial Networks (GANs) or Transformers, to generate outputs that mimic human-created content.
The potential impact of Gen AI on society is vast and multifaceted:
1. Creativity and Innovation: Gen AI can assist artists, writers, and musicians by generating ideas, drafts, or even complete works. This could lead to new forms of artistic expression and innovation.
2. Content Creation: Gen AI can automate the creation of content for various industries, such as advertising, journalism, and entertainment. This could increase efficiency and productivity but may also raise concerns about job displacement.


3. Personalization: Gen AI can create personalized content tailored to individual preferences, enhancing user experiences in areas like marketing, education, and entertainment.
4. Ethical and Legal Challenges: Gen AI raises questions about authorship, intellectual property, and the potential for misuse, such as creating deepfakes or spreading misinformation.
5. Dependence on AI: As Gen AI becomes more sophisticated, there's a risk of over-reliance on AI-generated content, which could stifle human creativity and critical thinking skills.
6. Economic Impact: Gen AI could lead to new industries and job opportunities, but it may also disrupt existing ones, leading to economic inequality if not managed properly.
7. Privacy Concerns: Gen AI models often require large amounts of data for training, which can raise privacy concerns if the data is sensitive or personal.
In conclusion, Gen AI has the potential to revolutionize various sectors of society, but it also presents significant challenges that need to be addressed through thoughtful regulation, ethical guidelines, and ongoing public discourse.

8. ADVANTAGES & DISADVANTAGES

Advantages

- Fast, natural-language responses
- Easy to use interface
- Free and accessible via browser

Disadvantages

- Depends on API availability
- Limited by token usage and response length
- No voice input or multilingual support (yet)

9. CONCLUSION

EduTutor AI serves as a practical, beginner-friendly AI tutoring app leveraging IBM Watsonx. It demonstrates the integration of cloud AI models with front-end frameworks like Streamlit.

10. FUTURE SCOPE

- Add voice input
- Expand to subject-specific modules
- Add multi-language support
- Use authentication for user tracking

11. APPENDIX

Source Code

```
import streamlit as st
from ibm_watsonx_ai.foundation_models import ModelInference

# Watsonx credentials and settings
model_id = "ibm/granite-3-8b-instruct"
project_id = "9d058698-f151-404a-af06-722f7cab493a"
credentials = {
   "url": "https://eu-de.ml.cloud.ibm.com",
   "apikey": "sCZynfN1FMBJSW-3waswn2krcNtypCQkqcCFJxOEcDku"
}

# Streamlit UI
st.title("EduTutor AI")

question = st.text_input("Ask your question:")

if st.button("Get Answer") and question.strip() != "":
   model = ModelInference(
      model_id=model_id,
      params={
         "decoding_method": "greedy",
         "max_new_tokens": 500
      },
      project_id=project_id,
```

```
    credentials=credentials
)

response = model.generate(question)
answer = response["results"][0]["generated_text"]
st.write(answer)
```

Dataset Link

https://www.ibm.com/docs/en/watsonx-as-a-service?topic=models-granite-13b-instruct

GitHub & Project Demo Link

GitHub Repo: https://github.com/CHARANRAVILLA/Edututor_AI