PAVENDADAR BHARATHIDASAN COLLEGE OF ARTS AND SCIENCE

PROJECT: EduTutor AI: Personalized Learning with Generative AI and LMS Integration

**Team ID :** NM2025TMID11254

**Team Size :** 4

**Team Leader :** Mathan K

**Team member :** Balamurugan M

**Team member :** Jegadeeswaran S

**Team member :** Sanjai S

Project Report: EduTutor AI

# INTRODUCTION

## 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.

## 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.

# IDEATION PHASE

## 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.

## 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."

## Brainstorming

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

# REQUIREMENT ANALYSIS

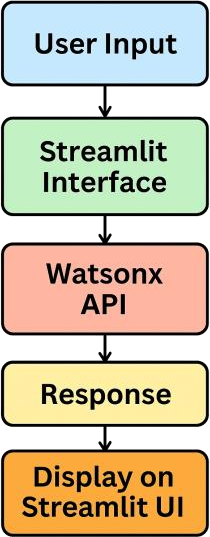
## Customer Journey Map

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

## Solution Requirement

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

## Data Flow Diagram

****

* 1. **Technology Stack**
     + Frontend: Streamlit
     + Backend: Python
     + AI Model: IBM Watsonx (Granite 3B-Instruct)
     + Hosting: Streamlit Cloud

# PROJECT DESIGN

## Problem Solution Fit

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

## Proposed Solution

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

## 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

# PROJECT PLANNING & SCHEDULING

* 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 |

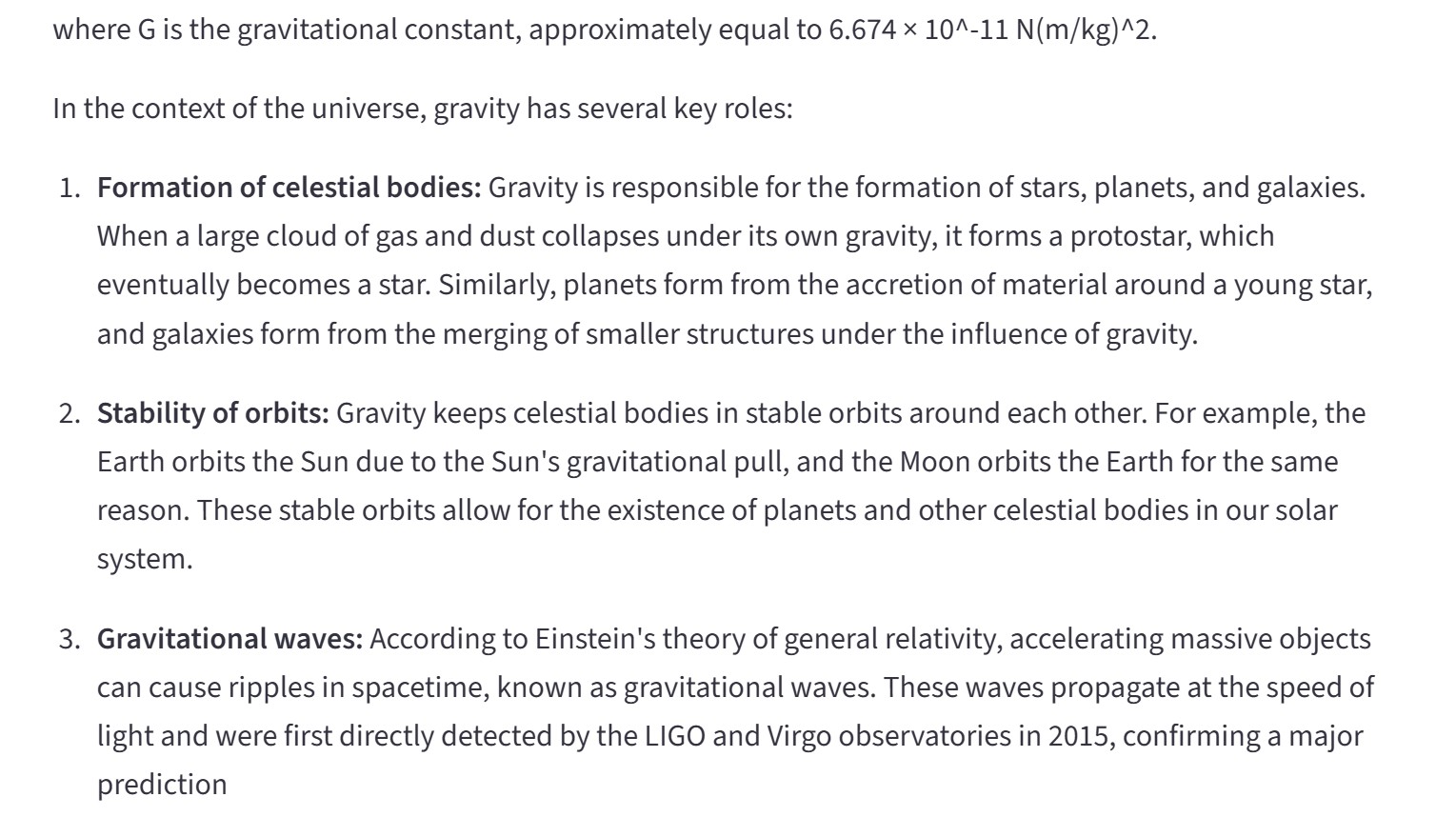
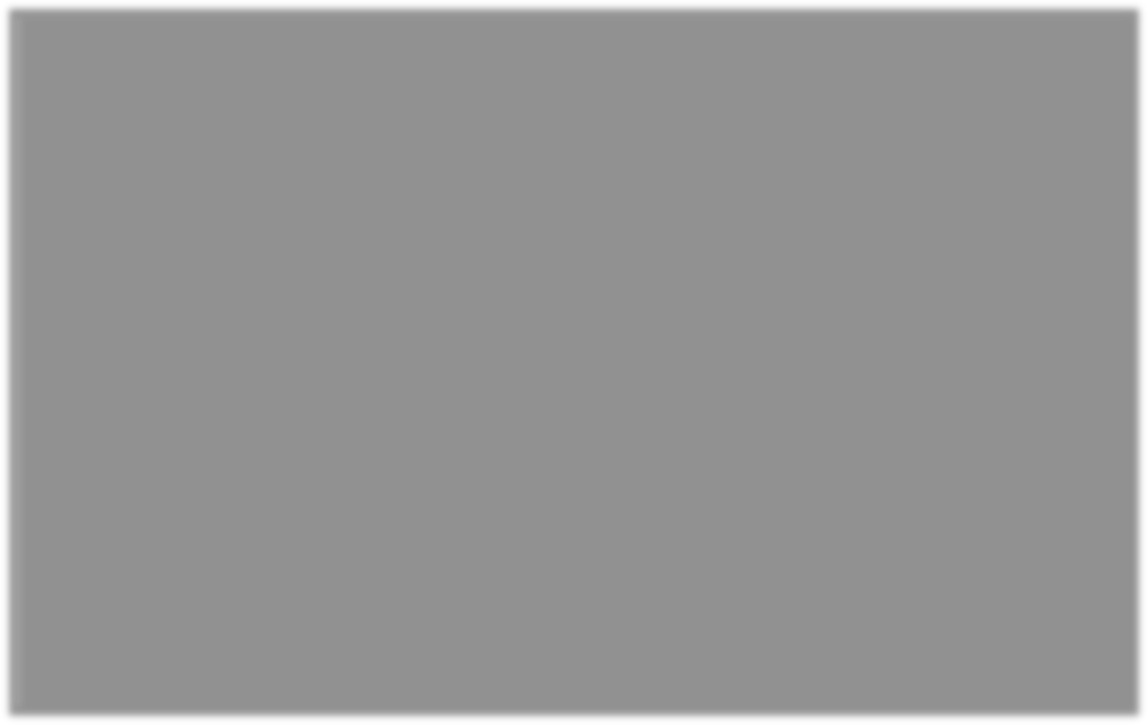
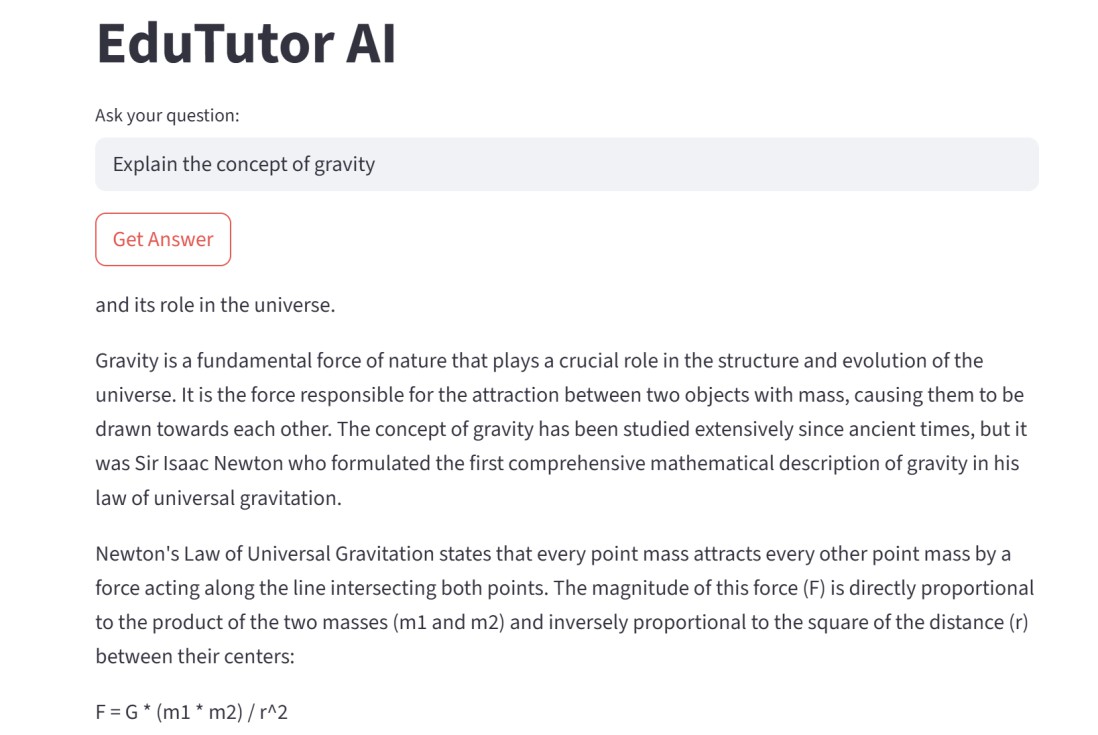
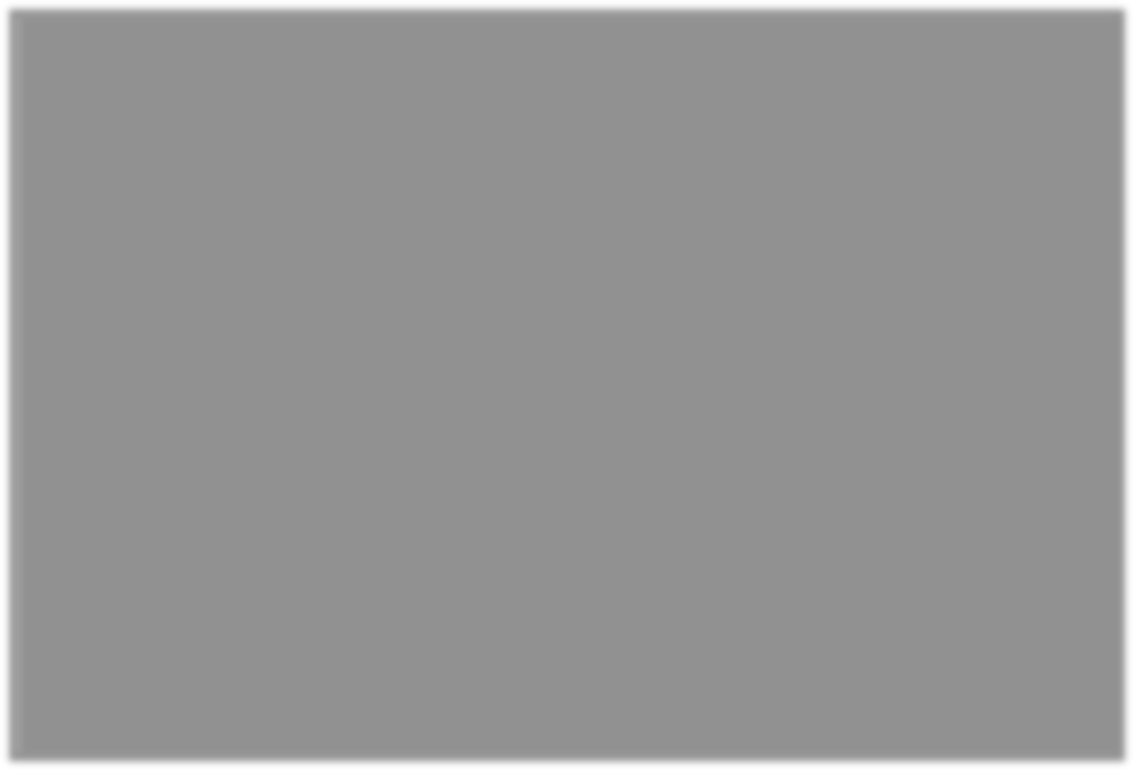
# FUNCTIONAL AND PERFORMANCE TESTING

## 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

# RESULTS



1. **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)

# 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.

# FUTURE SCOPE

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

# 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 = "e148ca84-35e1-433d-9d5e-a71c64c3def8" credentials = {

"url": "https://eu-de.ml.cloud.ibm.com",

"apikey": "bCDB66qGQ4GEdDAu6o6kQ-BM4iLenxHfXZDZCrwtMwKf"

}

# 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)