CAPSTONE PROJECT

PROJECT TITLE

Presented By:
1. Anushka Sharma-MIT AOE Pune-Btech ENTC

edunet foundation

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

A College Admission Agent, powered by RAG (Retrieval-Augmented Generation), streamlines the student admission process. It retrieves and summarizes admission policies, eligibility criteria, and FAQs from institutional databases and official sources. Prospective students can ask natural language questions and receive accurate, up-to-date responses instantly. The agent helps with course selection, application guidance, fee structure, and important deadlines. Using trusted, real-time data, it reduces manual inquiries and enhances applicant experience. This AI-driven assistant boosts transparency, accessibility, and efficiency in college admissions

PROCESSES, AND SEAT AVAILABILITY.

1. Data Collection:
Aggregate admission-related documents such as policies, eligibility criteria, course catalogs, FAQs, and fee structures from institutional databases, official college websites, and government education portals.
Integrate with APIs or authorized data repositories for real-time access to updates on admission dates, application processes, and seat availability.

2. Data Preprocessing:
Clean and structure unorganized or semi-structured documents using NLP techniques like sentence segmentation and named entity recognition (NER).
Standardize formats across different data sources for uniform retrieval and ensure removal of outdated or duplicate content.

3. Retrieval Module:
Implement a vector-based document retrieval system (e.g., FAISS, Pinecone) to enable fast and relevant extraction of information based on user queries.
Use semantic search techniques (e.g., Dense Passage Retrieval or BM25) to improve accuracy and relevance of the retrieved documents.

PROCESSES, AND SEAT AVAILABILITY.

4. Generation Module (RAG):
Use a pre-trained transformer model (e.g., BERT or GPT) integrated with the retriever to generate coherent, context-aware answers.
Ensure responses are grounded in retrieved documents to maintain factual accuracy and avoid hallucinations.

5. Natural Language Interface:
Design a chatbot/web interface that allows students to interact in natural language.
Include multilingual support and user personalization features (e.g., saving previous queries, suggesting relevant information).

6. Deployment:
Host the solution on a scalable cloud platform (e.g., IBM Cloud, AWS, or Azure) with proper data privacy and security compliance.
Ensure high availability and low latency response time for seamless user interaction.

PROCESSES, AND SEAT AVAILABILITY.

7. Evaluation:
Measure the system's performance using metrics such as F1-score for retrieval accuracy, BLEU or ROUGE for generation quality, and user satisfaction surveys.
Continuously refine based on feedback, usage logs, and changing institutional guidelines.

Result:
The AI-powered admission assistant will significantly reduce manual workload on administrative staff, provide prospective students with instant and reliable information, and increase overall satisfaction with the admission process. By leveraging real-time data and advanced NLP, it builds a smarter, faster, and more transparent college admission system.

SYSTEM APPROACH

The "System Approach" section outlines the strategy and methodology for building and deploying the AI-powered college admission assistant. This system leverages NLP, retrieval mechanisms, and transformer-based language models to provide real-time, factual responses to student queries. The approach consists of the following elements:

- **1. System Requirements**
- **Hardware Requirements:**
- Minimum 8 GB RAM (16 GB recommended for training)
- 4-core CPU (or GPU-enabled system for faster inference)
- 100 GB storage (for institutional documents and vector databases)
- **Software Requirements:**
- Operating System: Windows/Linux/MacOS
- Python 3.8 or later
- Git (for version control)
- Web browser (for chatbot interface testing)

SYSTEM APPROACH

- **1.Cloud Infrastructure (optional but recommended):**
- IBM Cloud / AWS / Azure for scalable deployment
- Docker (for containerization)
- MongoDB or Firebase (for user data storage)
  2. Libraries Required to Build the Model
- **A. Data Handling and Search**
- **pandas** – Used to work with and clean data.
- **numpy** – Helps with mathematical operations and numbers.
- **beautifulsoup4** or **requests** – Used to collect data from websites (if allowed).
- **langchain** – Connects the chatbot with AI models using the RAG method.
- **faiss** or **pinecone-client** – Helps the system search answers based on meaning, not just keywords.
- **sentence-transformers** – Turns text into numbers (embeddings) so it can be searched easily.

SYSTEM APPROACH

- 3. Natural Language Understanding
- **transformers** – Gives access to powerful AI models like GPT and BERT.
- **nltk** or **spaCy** – Used to break text into words, find names, and clean text.
- **torch** or **tensorflow** – These are tools that run the AI models in the background.
- 4. User Interface (optional)
- **streamlit** or **Flask** – Used to build the chatbot webpage or interface.
- **gradio** – Makes a simple and interactive interface with very little code.
  5. Deployment and Monitoring
- **gunicorn** or **uvicorn** – Used to run the chatbot app on a server.
- **Docker** – Packs the whole project so it can run easily anywhere.
- **Prometheus** or **Grafana** – Optional tools to check performance and logs of the chatbot.

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**
- The project employs a **Retrieval-Augmented Generation (RAG)** framework that combines two core components:
- **Retriever**: A dense vector-based retriever (e.g., FAISS or Pinecone) that searches and retrieves the most relevant documents or passages based on the user's query.
- **Generator**: A transformer-based language model (e.g., **GPT-3.5**, **BERT**, or **FLAN-T5**) that uses the retrieved documents to generate accurate and context-aware responses.
- **Justification for RAG:**
- College admission queries require **factually accurate** and **context-specific** answers.
- Static LLMs may hallucinate or provide outdated information.
- RAG ensures grounding of generated responses in **real-time, trusted institutional data**, improving both accuracy and reliability.

ALGORITHM & DEPLOYMENT

- **Data Input:**
- The system uses the following input features:
- **User Query** (natural language question from the student)
- **Institutional Documents**, including:
  - Admission policies
  - Eligibility criteria
  - Course details and descriptions
  - Fee structures
  - FAQs and important deadlines
- **Optional Enhancements:**
- Student's selected stream or interest area
- Location/language preference for personalized results

ALGORITHM & DEPLOYMENT

Training Process:

The retriever uses sentence embeddings (from models like Sentence-BERT) to map queries and documents into vector space.

A pre-trained generator model is used and optionally fine-tuned on a dataset of admission-related Q&A.

Techniques like cross-validation, human evaluation, and similarity scoring are used to improve performance.

The retriever may be improved using techniques like contrastive learning or triplet loss.

Prediction Process:

The student enters a query.

The system retrieves the most relevant documents using the retriever.

The generator model uses these documents to generate an accurate, contextual answer.

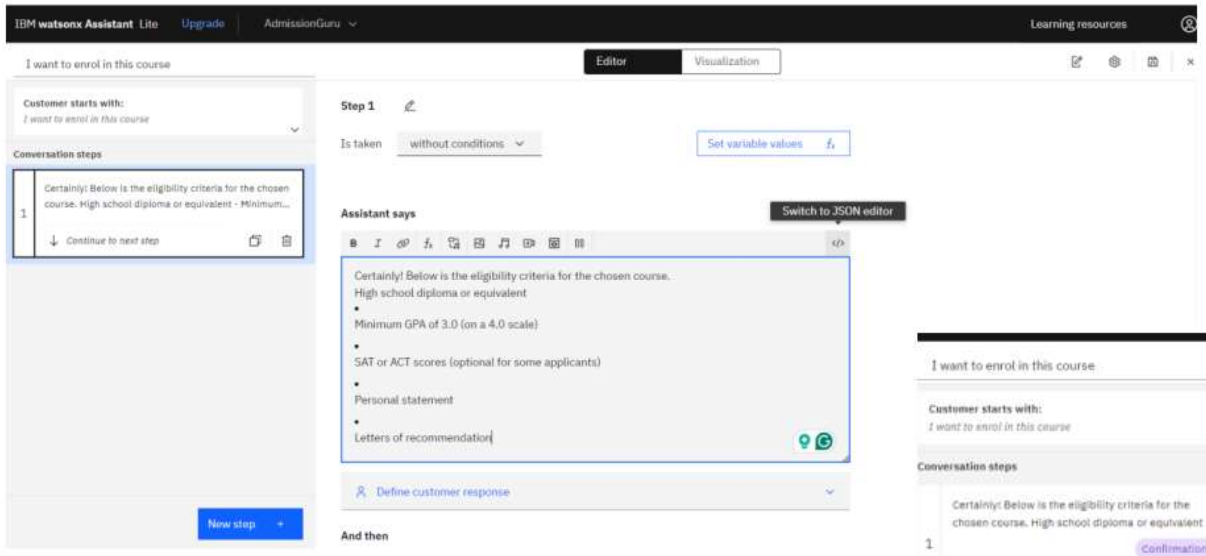
The result is shown to the user in real time, along with optional source references.

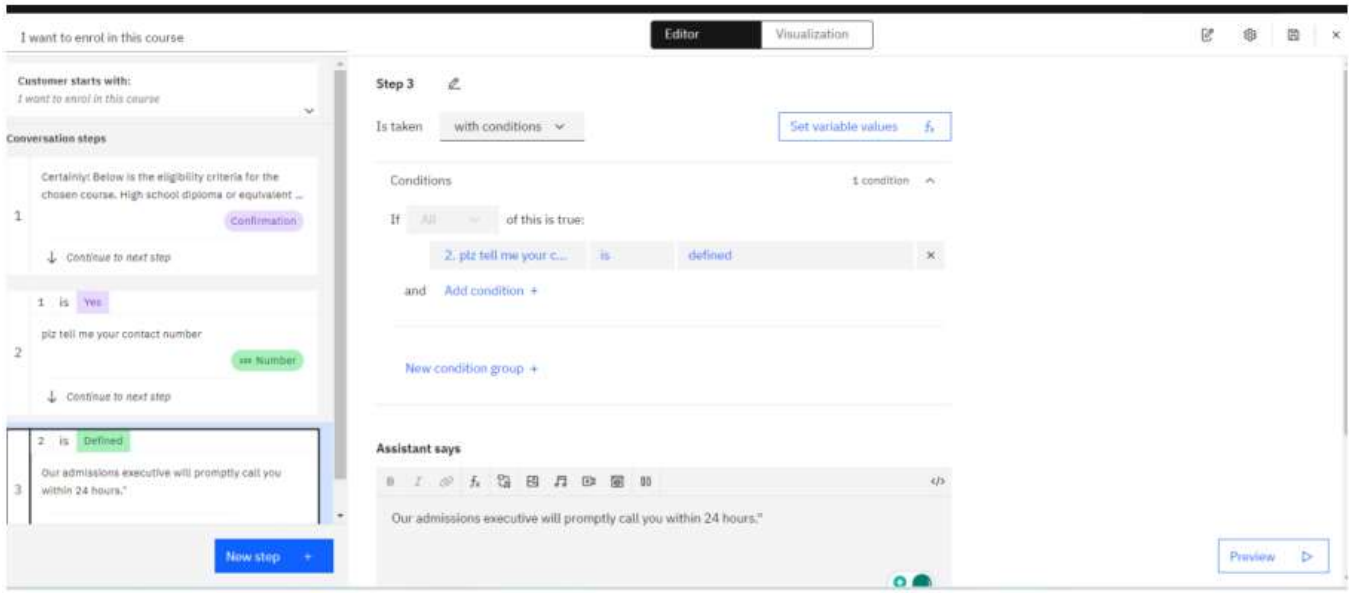
The system always works with the latest documents, so there's no need to retrain the model each time policies are updated.

Real-Time Consideration:

Documents and policies can be updated in real time in the backend corpus, so the system always works on the latest information—without retraining the model.

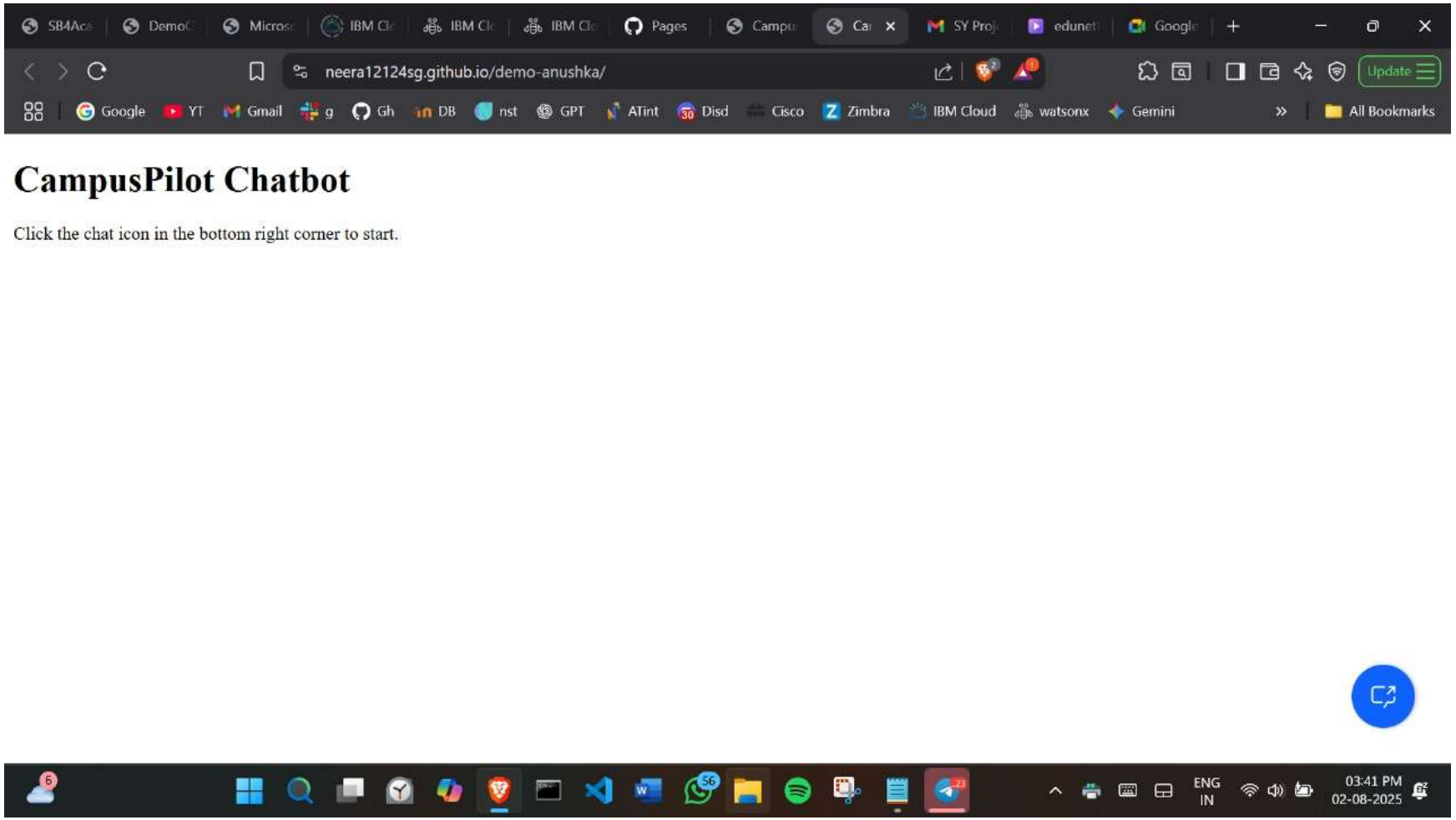
ALGORITHM & DEPLOYMENT



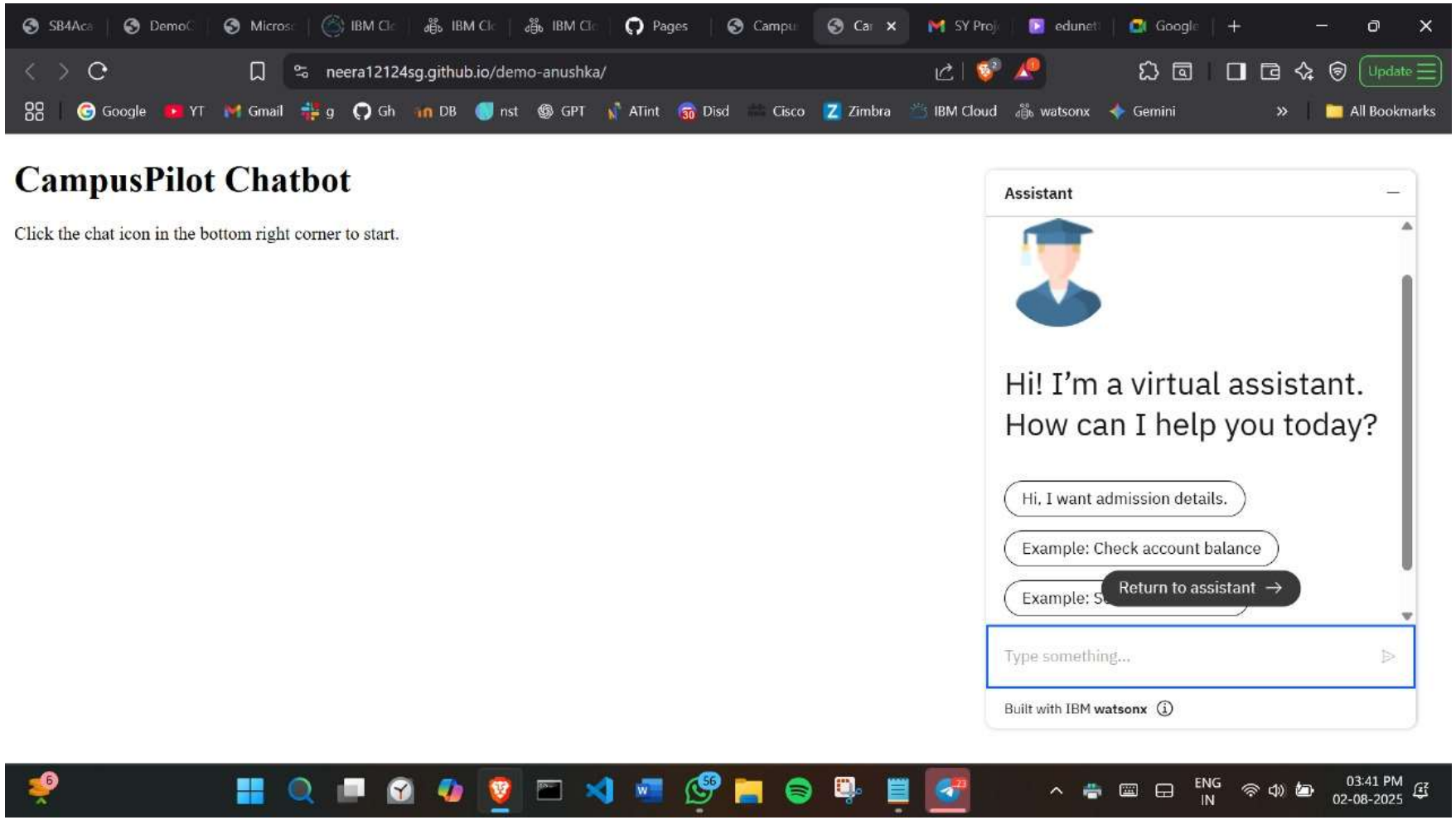


edunet foundation

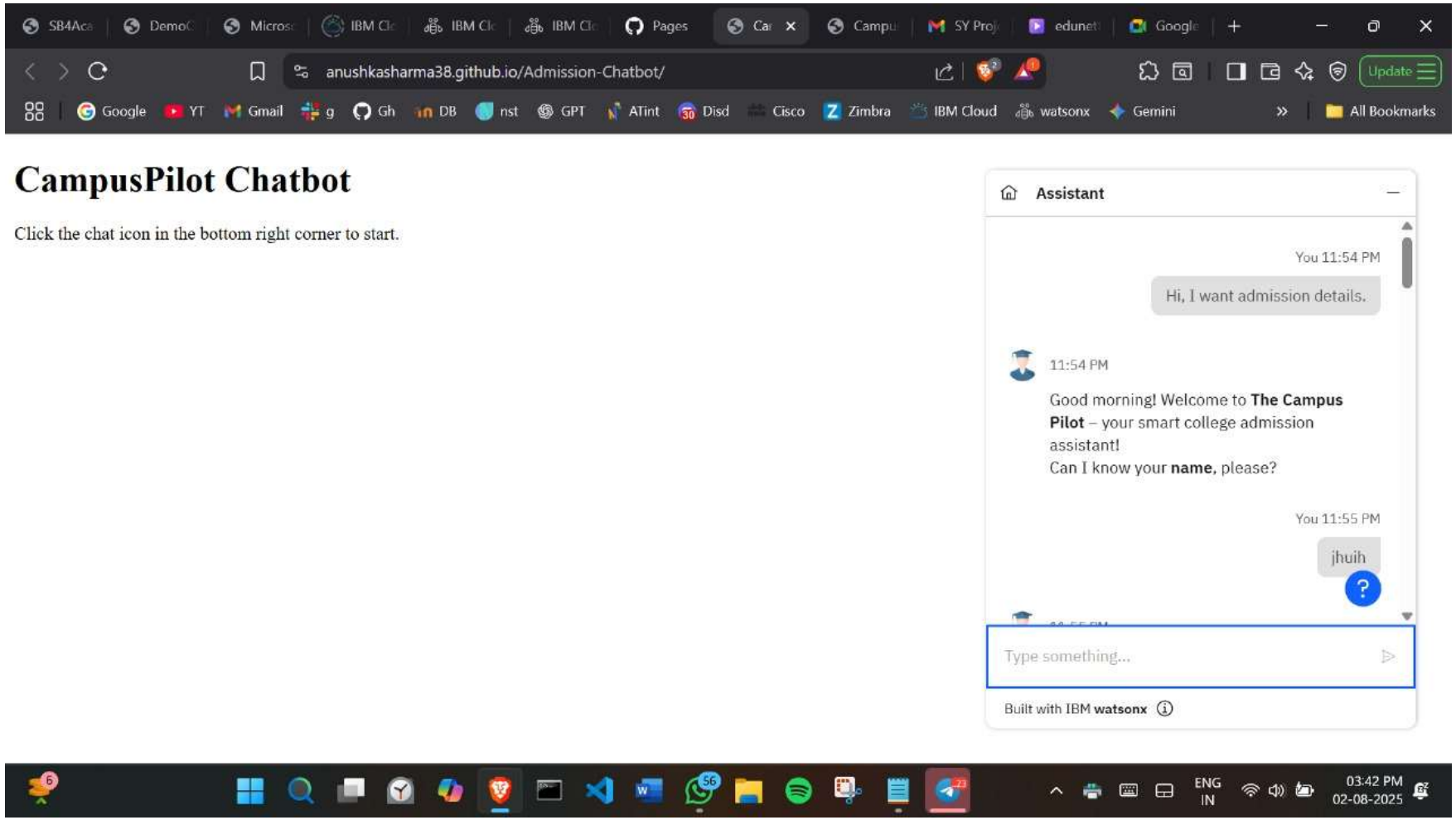
RESULT


neera12124sg.github.io/demo-anushka/
CampusPilot Chatbot
Click the chat icon in the bottom right corner to start.

RESULT

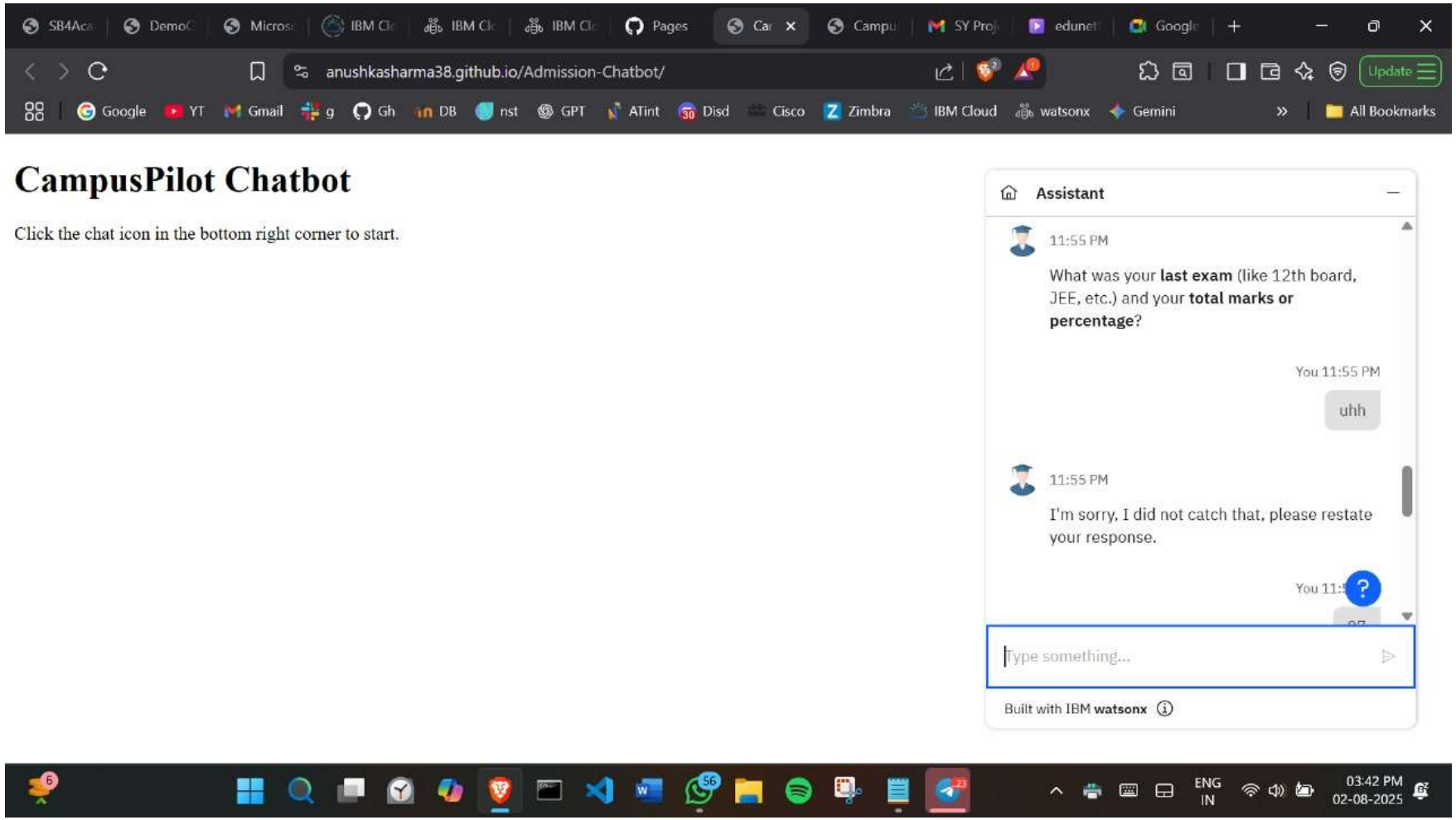

neera12124sg.github.io/demo-anushka/
CampusPilot Chatbot
Click the chat icon in the bottom right corner to start.
Assistant
Hi! I'm a virtual assistant.
How can I help you today?
Hi, I want admission details.
Example: Check account balance
Return to assistant
Type something...
Built with IBM watsonx

RESULT

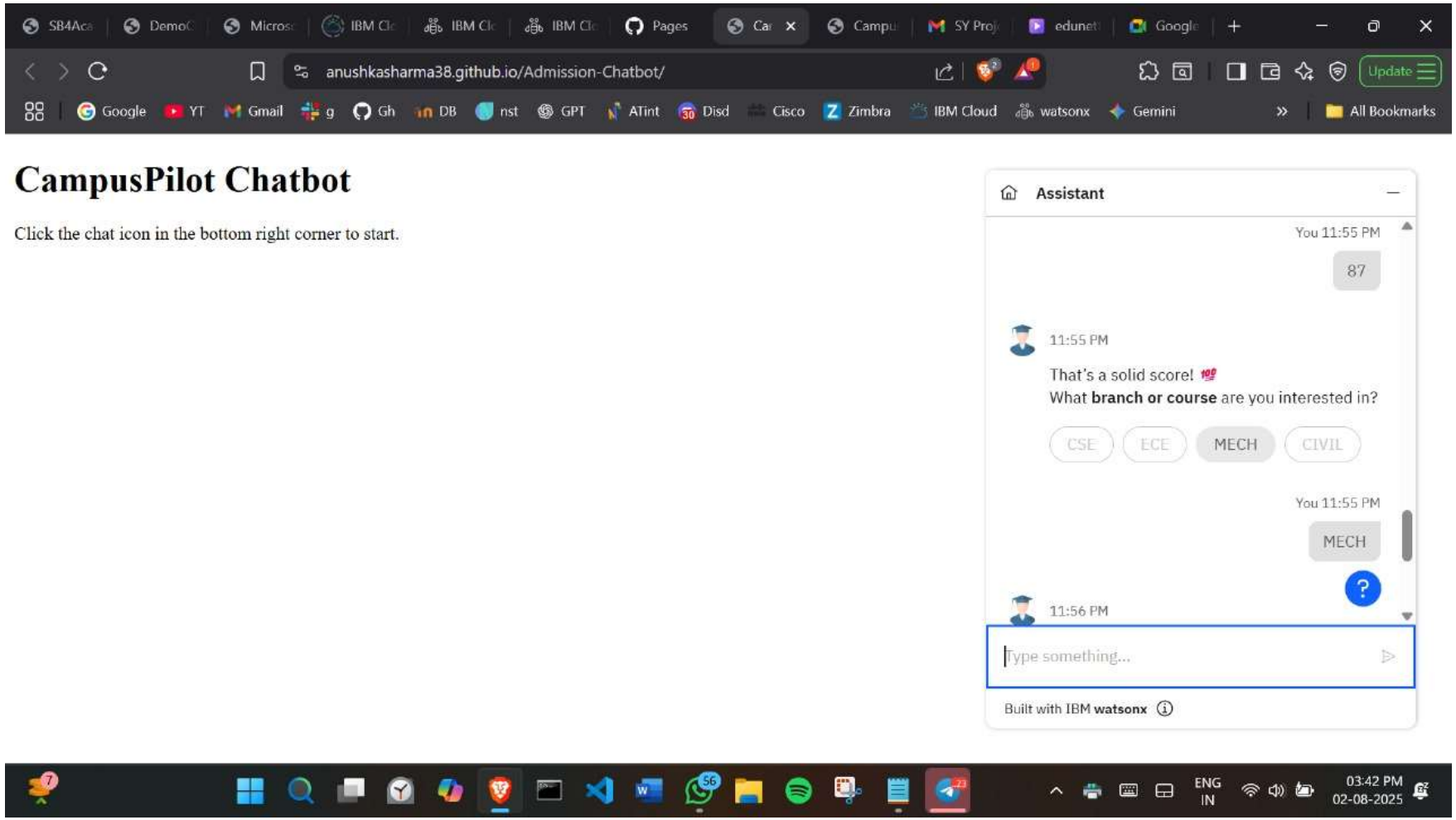


edunet foundation

RESULT

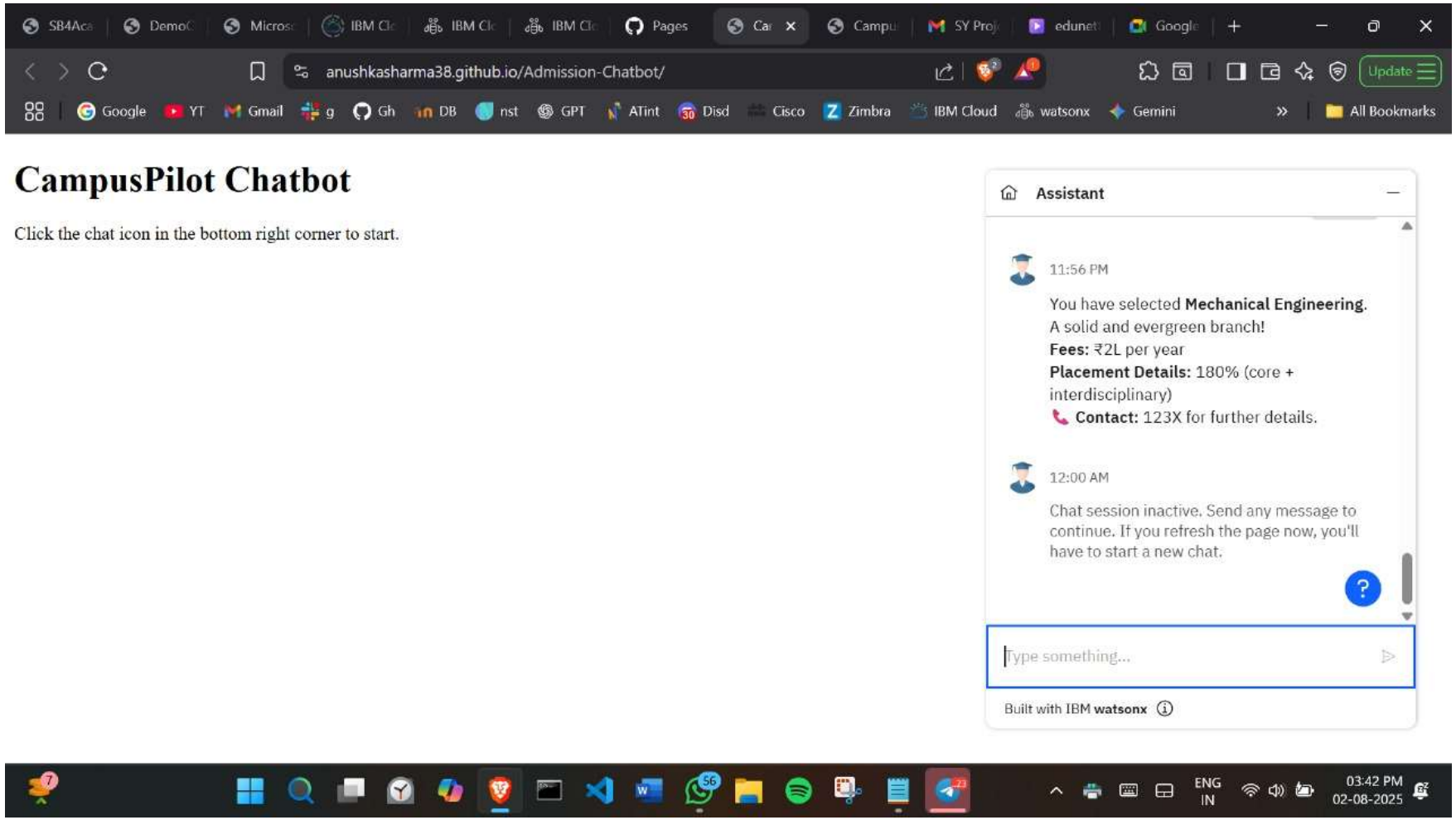

anushkasharma38.github.io/Admission-Chatbot/
CampusPilot Chatbot
Click the chat icon in the bottom right corner to start.
Assistant
11:55 PM
What was your last exam (like 12th board, JEE, etc.) and your total marks or percentage?
You 11:55 PM
uhh
11:55 PM
I'm sorry, I did not catch that, please restate your response.
Type something...
Built with IBM watsonx
ENG IN
03:42 PM
02-08-2025

RESULT



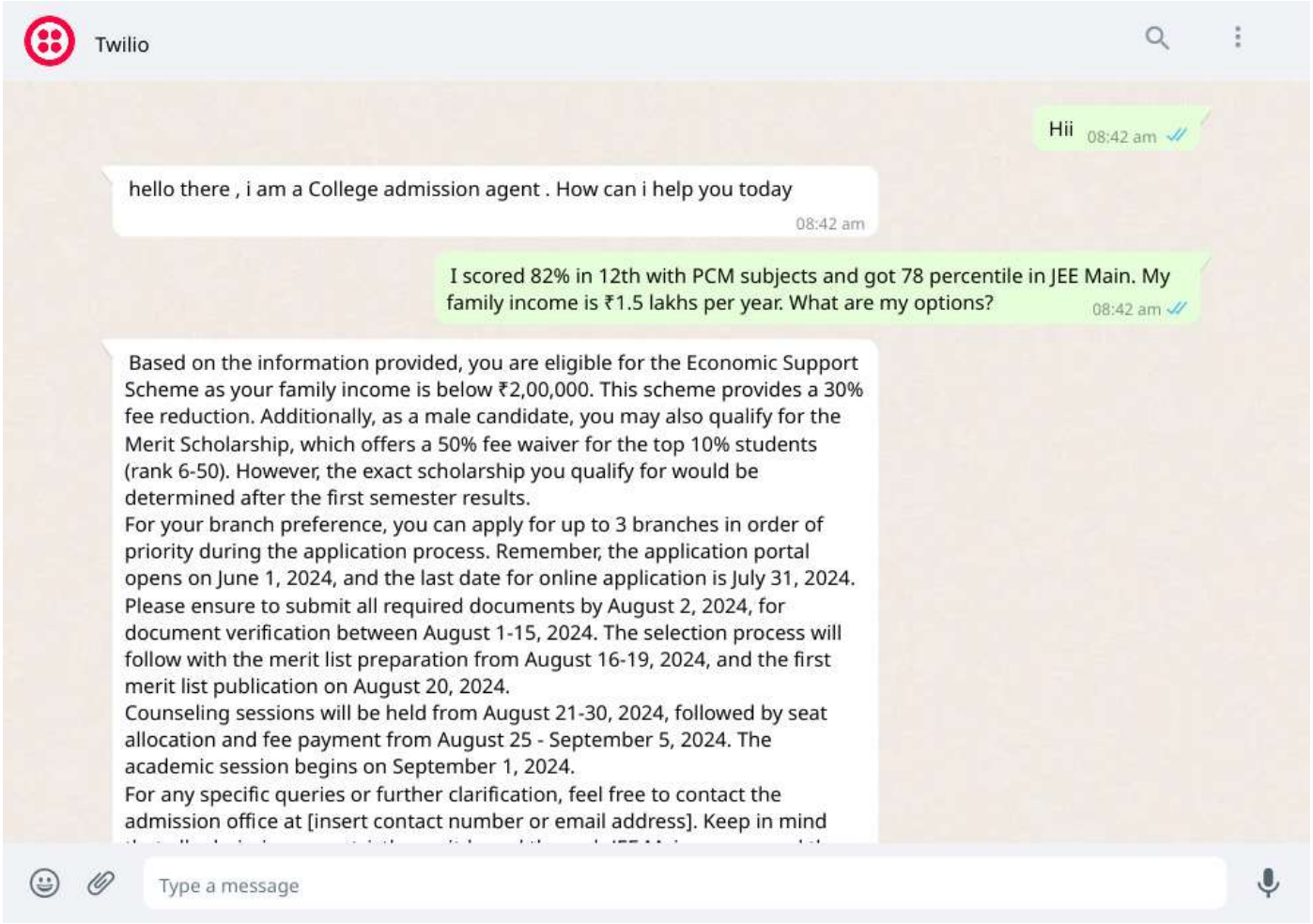
edunet foundation

RESULT



edunet foundation

RESULT


Twilio
Hii 08:42 am
hello there , i am a College admission agent . How can i help you today 08:42 am
I scored 82% in 12th with PCM subjects and got 78 percentile in JEE Main. My family income is ₹1.5 lakhs per year. What are my options? 08:42 am
Based on the information provided, you are eligible for the Economic Support Scheme as your family income is below ₹2,00,000. This scheme provides a 30% fee reduction. Additionally, as a male candidate, you may also qualify for the Merit Scholarship, which offers a 50% fee waiver for the top 10% students (rank 6-50). However, the exact scholarship you qualify for would be determined after the first semester results.
For your branch preference, you can apply for up to 3 branches in order of priority during the application process. Remember, the application portal opens on June 1, 2024, and the last date for online application is July 31, 2024.
Please ensure to submit all required documents by August 2, 2024, for document verification between August 1-15, 2024. The selection process will follow with the merit list preparation from August 16-19, 2024, and the first merit list publication on August 20, 2024.
Counseling sessions will be held from August 21-30, 2024, followed by seat allocation and fee payment from August 25 - September 5, 2024. The academic session begins on September 1, 2024.
For any specific queries or further clarification, feel free to contact the admission office at [insert contact number or email address]. Keep in mind
Type a message

CONCLUSION

- The proposed College Admission Agent using RAG proved to be highly effective in delivering fast, accurate, and reliable responses to student queries. It successfully reduced manual workload and improved user experience with real-time guidance on courses, eligibility, fees, and deadlines.
- **Challenges Faced:**
- Integrating multiple data sources with different formats
- Ensuring factual accuracy to prevent misinformation
- Handling vague or ambiguous user queries
- **Potential Improvements:**
- Add voice input and multilingual support
- Include a recommendation system for course selection
- Integrate with college application portals for end-to-end assistance
- **Impact:**
  This AI solution improves transparency, saves time, and ensures that students receive correct information instantly, enhancing the overall admission process.

FUTURE SCOPE

- **Expand Data Sources**
  - Integrate real-time data from government portals, ranking websites, and social media to improve accuracy and relevance.
- **Algorithm Optimization**
  - Fine-tune LLMs with domain-specific datasets
  - Use hybrid retrieval (dense + sparse) to improve response precision
- **Scalability Across Regions**
- Extend the system to support multiple colleges, cities, or states
- Add language localization for regional outreach
- **Emerging Technology Integration**
- Use **edge computing** to reduce latency in rural or low-connectivity areas
- Incorporate **advanced ML models** like multi-modal agents for document + voice processing
- **User Personalization**
- Recommend colleges/courses based on student interests and past queries
- Store preferences for faster future responses

REFERENCES

•IBM Skills Network. *Retrieval-Augmented Generation (RAG) Fundamentals*. IBM Online Learning Platform. https://skills.network

•IBM Cloud. *AI-Powered Virtual Assistants for Business Applications*. IBM Cloud Docs. https://cloud.ibm.com/docs

Course PDFs, slide decks, and session notes from IBM classes attended (internal use).

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence
IBM SkillsBuild

IBM

ANUSHKA SHARMA

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 16, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/4dfb96ec-9657-4a86-8728-cb459414d430

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM

ANUSHKA SHARMA

Has successfully satisfied the requirements for:

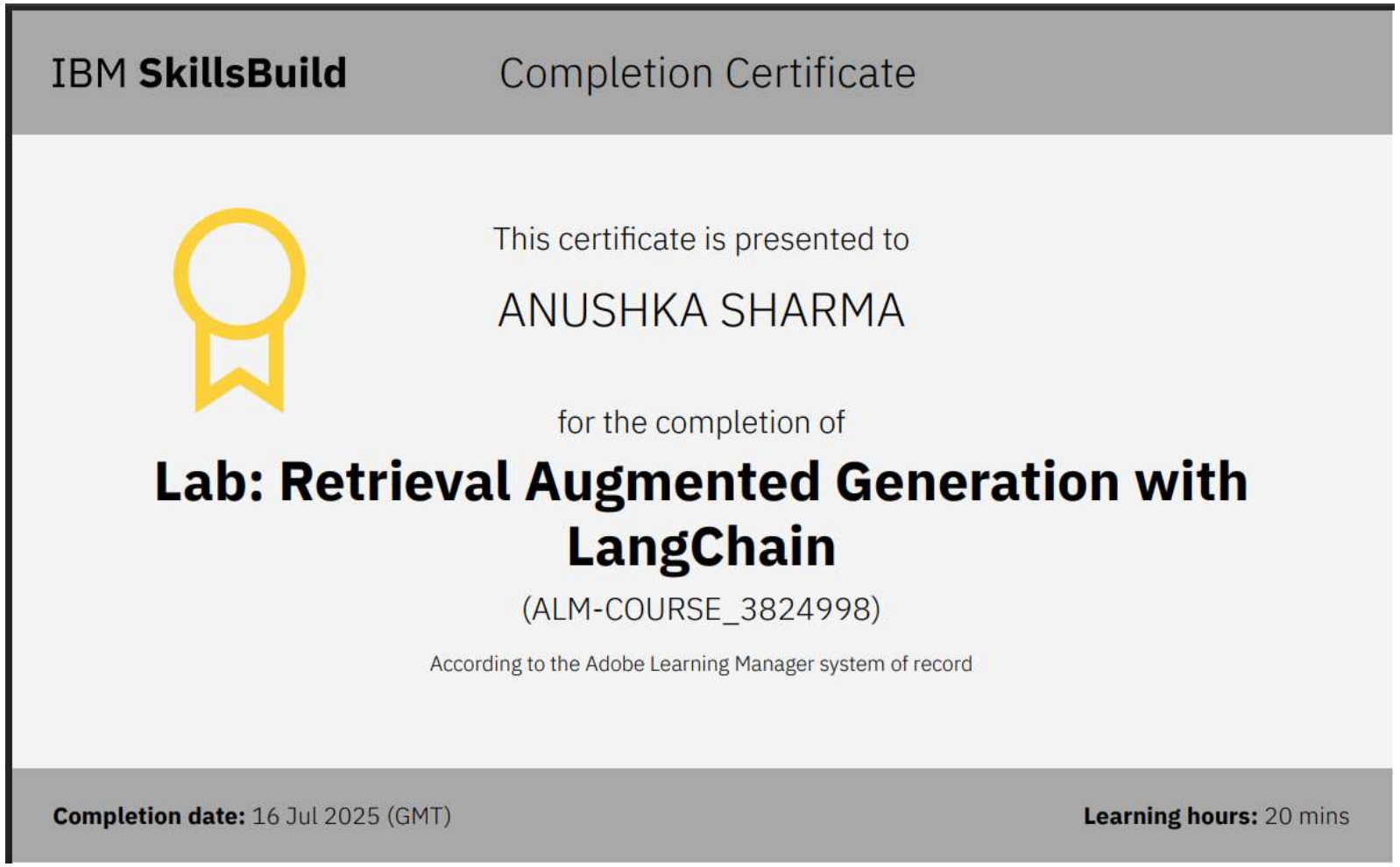
Journey to Cloud: Envisioning Your Solution

Issued on: Jul 16, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/568d5f28-9ebd-4544-8a2d-bb7c338a338a

IBM

IBM CERTIFICATIONS



IBM **SkillsBuild** Completion Certificate

This certificate is presented to

ANUSHKA SHARMA

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 16 Jul 2025 (GMT) **Learning hours:** 20 mins

THANK YOU

edunet foundation