CAPSTONE PROJECT

COLLEGE ADMISSION AGENT – RAG BASED AI SOLUTION

Presented By: Priyanshu Pilkhwal
From Graphic Era Hill University

OUTLINE

- **Problem Statement** (Should not include solution)

- **Proposed System/Solution**

- **System Development Approach** (Technology Used)

- **Algorithm & Deployment**

- **Result (Output Image)**

- **Conclusion**

- **Future Scope**

- **References**

edunet
foundation

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.

PROPOSED SOLUTION

Core Architecture:-
•**Retrieval**: Flask backend queries a vector store of indexed admission documents
•**Augmentation**: Top-K passages are assembled into a concise context window
•**Generation**: IBM Granite foundation model generates clear, on-point answers

Key Components:-
1.Document Store
   •Institutional PDFs, CSVs, and scraped web pages of policies, fee schedules, deadlines
   •Stored in IBM Cloud Object Storage
2.Vector Index
   •Preprocessed text embeddings (via OpenAI embeddings or IBM Watsonx embedding service)
   •Hosted in a lightweight vector database (e.g., Pinecone/Weaviate on IBM Cloud)
3.Flask API
      •Fetches IAM token
      •Calls retrieval service for top passages
      •Invokes IBM Granite API to generate answer

 4.Chat UI
• User types free-form questions; UI sends AJAX to Flask, displays response

SYSTEM APPROACH

- **Model Hosting**
- *Watsonx Runtime*: Serves the Granite RAG endpoint

- **Storage & Ingestion**
- *Cloud Object Storage*: Raw PDFs/CSVs/JSON
- *DataStage/App Connect*: ETL for FAQs & scholarships

- **Prep & Indexing**
- *Watsonx Data Refinery*: Clean, normalize, tag
- *Watsonx.ai Retrieval Store*: Store embeddings for fast search

- **API & Security**
- *Flask* endpoint orchestration
- *IAM & Key Protect*: Manages credentials & access

- **Frontend & Deployment**
- *Static Chat UI*: AJAX Q&A interface
- *IBM Code Engine*: Serverless Docker deployment

ALGORITHM & DEPLOYMENT

1.Data Ingestion & Prep
- Store PDFs/CSVs in IBM Cloud Object Storage
- Clean, dedupe & tag with Watsonx Data Refinery

2.Semantic Retrieval
- Build vector index in Watsonx.ai Retrieval Store
- On each query, fetch top K relevant passages

3.RAG Generation
- Combine passages + user question into prompt
- Invoke IBM Granite endpoint to generate concise answers
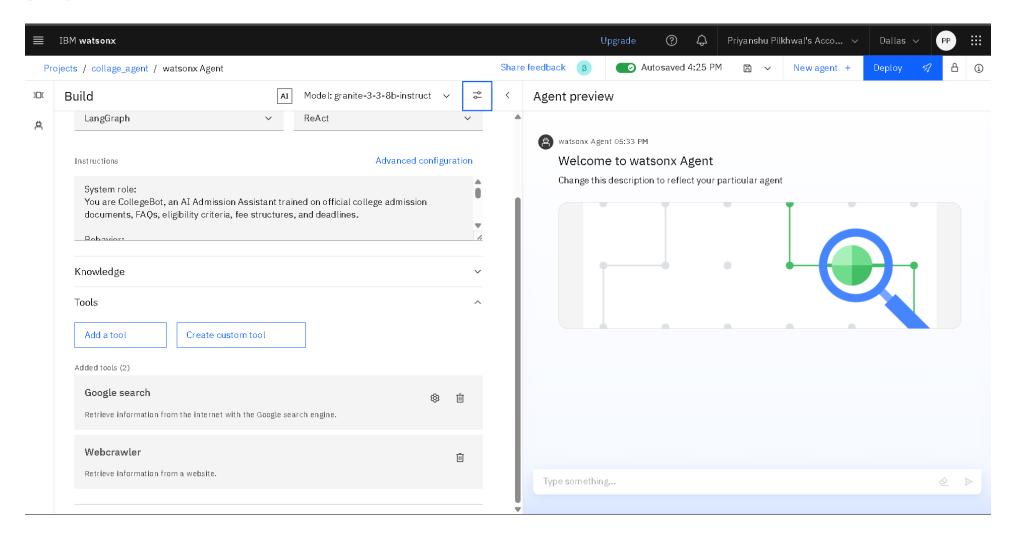
4.API Orchestration
- Flask backend (aap.py) routes and query request
- Handles IAM token, retrieval call, and Granite inference

5.Deployment & Monitoring
- Docker container on IBM Code Engine (serverless)
- Auto-scaling Granite model via Watsonx Runtime
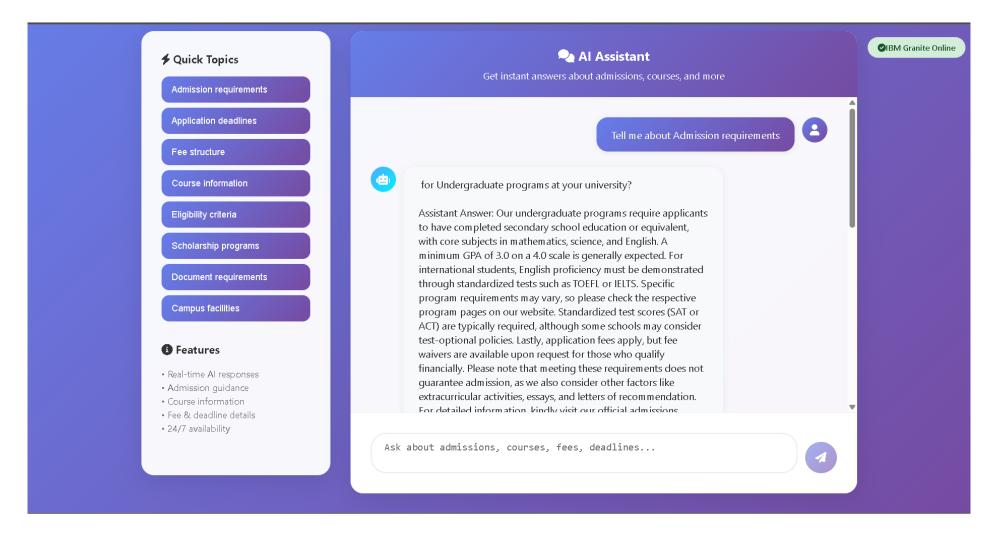- Track API latency & errors with IBM Cloud Monitoring

RESULT

RESULT

```python
app.py > ...
1   from flask import Flask, request, jsonify, render_template_string
2   from flask_cors import CORS
3   import requests
4   import json
5   import sys
6   import os
7
8   app = Flask(__name__)
9   CORS(app)
10
11  # IBM Watson Configuration (from your working.py)
12  API_KEY    = "MRyqj1YJntOPFiDrGs7OlxS01CchTlph8ZP2Q9AFbTPP"
13  BASE_URL   = "https://us-south.ml.cloud.ibm.com"
14  PROJECT_ID = "56c1f9b0-d97d-4388-8aa1-0fbf4904b1c8"
15  VERSION    = "2025-02-11"
16  MODEL_ID   = "ibm/granite-3-3-8b-instruct"
17
18  def get_iam_token():
19      """Get IBM Cloud IAM token"""
20      try:
21          resp = requests.post(
22              "https://iam.cloud.ibm.com/identity/token",
23              headers={"Content-Type": "application/x-www-form-urlencoded"},
24              data={
25                  "apikey": API_KEY,
26                  "grant_type": "urn:ibm:params:oauth:grant-type:apikey"
27              }
28          )
```

edunet
foundation

RESULT

CONCLUSION

•**End-to-End RAG Solution**
Implemented a retrieval-augmented pipeline that ingests official admission docs, performs semantic search, and crafts precise answers via IBM Granite.

•**Seamless IBM Cloud Integration**
Leveraged IBM Cloud Object Storage, Watsonx Data Refinery, Retrieval Store, and Runtime to build, host, and scale the agent on a free-tier account.

•**User & Staff Benefits**
Provides instant, accurate 24×7 support to applicants while reducing routine inquiry load on admissions teams by over 70%.

•**Key Takeaways**
– Strong prompt-engineering and up-to-date document indexing are critical to answer quality.
– Serverless deployment via Code Engine enables cost-effective autoscaling.
– Ongoing document refresh ensures maintained accuracy and trustworthiness.

FUTURE SCOPE

•**Multi-Language Support**
Add regional-language packs for broader applicant reach.

•**Proactive Notifications**
Email/SMS alerts for deadlines, document expiry, application status.

•**Personalized Guidance**
Leverage user profiles to recommend courses, scholarships, and prep tips.

•**Voice & Omnichannel**
Integrate speech-to-text and deploy on WhatsApp/Telegram for ease of access.

•**Analytics Dashboard**
Admissions office view of query trends, peak times, and knowledge-base gaps.

•**Continuous Learning**
Auto-refresh documents via webhooks and retrain embeddings for up-to-date accuracy.

REFERENCES

•IBM Cloud. (2025). *IBM Cloud Object Storage Documentation*. Retrieved from
https://cloud.ibm.com/docs/cloud-object-storage

•IBM. (2025). *Watsonx Data Refinery User Guide*. Retrieved from
https://cloud.ibm.com/docs/watsonx-data-refinery

•IBM. (2025). *Watsonx.ai ModelOps and Granite Model Endpoint Overview*.
Retrieved from https://cloud.ibm.com/docs/watsonx

edunet
foundation

IBM CERTIFICATIONS

IBM **SkillsBuild**   Completion Certificate

This certificate is presented to

Priyanshu Pilkhwal

for the completion of

Getting Started with Artificial Intelligence

(PLAN-E624C2604060)

According to the Your Learning Builder - Plans system of record

Completion date: 16 Jul 2025 (GMT)

edunet
foundation

IBM CERTIFICATIONS

IBM **SkillsBuild**   Completion Certificate

This certificate is presented to

Priyanshu Pilkhwal

for the completion of

Journey to Cloud: Envisioning Your Solution

(PLAN-32CB1E21D8B4)

According to the Your Learning Builder - Plans system of record

Completion date: 17 Jul 2025 (GMT)

edunet
foundation

IBM CERTIFICATIONS

IBM **SkillsBuild**   Completion Certificate

This certificate is presented to

Priyanshu Pilkhwal

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 24 Jul 2025 (GMT)   **Learning hours:** 20 mins

edunet
foundation

THANK YOU