# RAG Implementation on Azure AI Foundry using Prompt Flow

This document outlines the work carried out by Vaishal Shah to implement Retrieval-Augmented Generation (RAG) using Azure AI Foundry's Prompt Flow capabilities. The objective of this task was to build a system that retrieves relevant context from indexed Azure Blob documents and uses Large Language Models (LLMs) to generate intelligent responses. The screenshots provided are evidence of successful execution and completion of this implementation.

## Overview of the Work

Vaishal integrated Azure AI Search with Prompt Flow to create a complete RAG pipeline. The process included the following components:

* - Azure Blob Indexing for ingesting and managing the document corpus.
* - Azure AI Search to perform semantic search over the indexed content.
* - GPT-4.1 model deployment in Azure OpenAI for response generation using chat API.
* - Prompt Flow with nodes connected to inputs, LLM, index lookup, and final output handler.

## Prompt Flow Setup

Below is the visual graph from Prompt Flow UI showcasing how the nodes are connected for index lookup and generating responses through the deployed GPT-4.1 model:

A screenshot of a computer

AI-generated content may be incorrect.

## Capabilities Demonstrated

The RAG flow successfully returned intelligent answers based on custom indexed content. This proves the following capabilities:

* - Effective use of Azure AI Search index for context retrieval.
* - Prompt Flow orchestration of LLM and data source retrieval.
* - Ability to customize prompt formats and logic.

## Skill Demonstration

The system successfully extracted resume insights and turned them into intelligent, context-aware responses. Here is the output showing LLM generating a context-aware joke based on a resume context:

A screenshot of a computer

AI-generated content may be incorrect.

## Azure AI Search Data Source Configuration

The system was connected to the 'vaishaltest' Azure AI Search resource, with an index configured to retrieve resume content. The data was securely stored in Azure Blob and processed through Azure AI Search and used as context input for the GPT-4.1 model. This setup highlights Vaishal’s ability to integrate various Azure components effectively to build a scalable and intelligent RAG pipeline.

## Conclusion

This project illustrates Vaishal Shah’s hands-on expertise in building and deploying end-to-end Retrieval-Augmented Generation (RAG) systems using Azure AI Foundry, Azure AI Search, Prompt Flow, and OpenAI LLMs. The demonstrated solution is both production-ready and extensible.