Azure AI Apps in a Day

Extend your app’s capabilities with Azure’s AI services

Meet our team

Cloud App and Integration (East Coast)

Ross Johnson
Practice Lead
ross.johnson@insight.com

Daniel Fang
Senior Architect
daniel.fang@insight.com

Rakesh Lakshminarayana
Architect
rakesh.lakshminarayana@insight.com

Content for today

Presentations:

- What are AI Apps?
- Building AI Apps & Techniques
- Azure’s Broad Set of AI Services
- Azure’s AI Supporting Capabilities
- RAG Pattern & Techniques
- Common Patterns and Architectures

Hands-on Labs

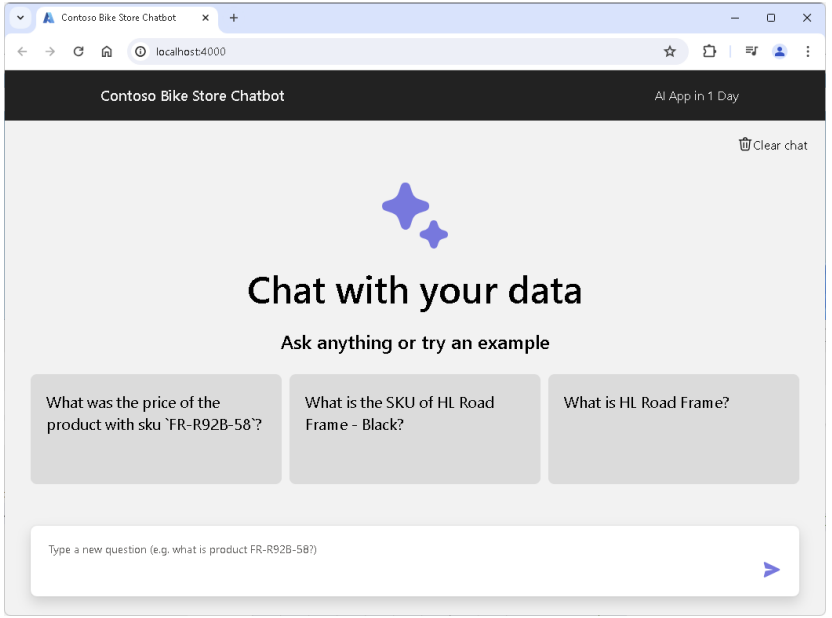
- Lab 1 - Interact with OpenAI Models
- Lab 2 - Building a Chatbot using RAG

Insight.

Hackathon

Get ready to build your very own Chatbot using RAG on Azure!

- Chatbot frontend (React)
- Restful API backend (node.js)
- Azure OpenAI GPT4o & Embedding
- Azure App Service
- Azure Cosmos DB for MongoDb
- Azure Deployment with Bicep
- Langchain (node.js)



Schedule

Time	Activity
08:30 (30m)	Presentation 1: AI Apps using Azure
09:00 (30m)	Tech check
09:30 (30m)	Morning break ☕
10:00 (2h)	Hands-on lab time (Lab 1)
12:00 (30m)	Lunch 🍕 🍔 🥗
12:30 (30m)	Presentation 2: RAG Pattern & Techniques
13:00 (15m)	Lab questions check-in
13:15 (2h)	Hands-on lab time (Lab 2)
15:15 (15m)	Day review and wrap

You are here

Insight.

Be ambitious.

AI Apps using Azure

What are AI Apps ?

Software applications that leverage AI technologies to carry out tasks traditionally requiring human intelligence. These tasks include learning from data, identifying patterns, making decisions, and solving complex problems

Natural Language Processing (NLP):
Enables apps to understand, interpret, and generate human language.

Machine Learning:
Algorithms that allow apps to learn from and adapt to new data over time.

Predictive Analytics:

Predictive analytics to forecast outcomes based on data

Computer Vision:
Allows apps to interpret and process visual information from the world.

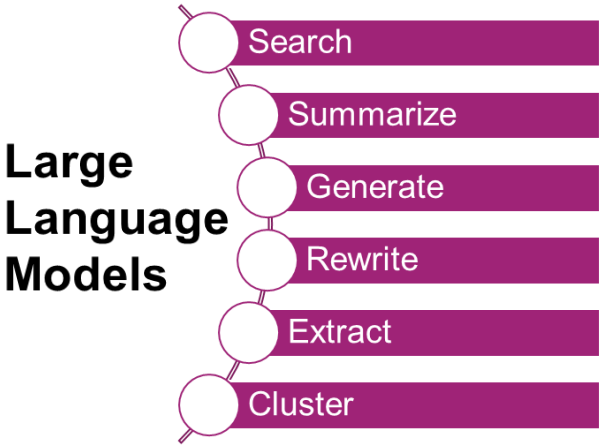
Automation:
Automates repetitive tasks, improving efficiency and accuracy.

Large Language Model (LLM)

Large Language Models are transforming the way application engage with users, enabling more natural and intelligent interactions. With LLMs, AI apps can comprehend, generate, and communicate in human language with remarkable complexity and precision.

Key Features:

- **Contextual Understanding**: Ability to grasp context and nuances in conversation, leading to more relevant and coherent responses.
- **Generative Abilities**: Create human-like text based on prompts, enabling applications like content creation and storytelling.
- **Adaptability**: Learn from diverse datasets, allowing for versatile applications across various domains.



Think broader, Generative AI

Generative AI creates new content by learning from existing data, enabling it to produce outputs like text, images, or music that mimic human creativity.

Chatbots and virtual assistants: Provide instant and accurate responses to customer queries

Content creation and design: Effective content creation and design to attract and engage customers

Accelerated automation: Boosts efficiency and drives greater productivity

Product and service innovation: Staying innovative and meeting evolving customer demands

Language translation and natural language processing: overcoming language barriers

Fraud detection and cybersecurity: Businesses face constant threats from fraudsters and cyberattacks

Predictive analytics and forecasting: For effective decision-making and operational efficiency

Medical research and diagnosis: potentially leading to faster and more accurate diagnoses

Building AI Apps

Creating a successful AI app requires careful attention to several key components and considerations to ensure it is effective, reliable, and user-friendly

- **Objectives and Use Case**: Define the specific problem your AI app will solve
- **AI Models**: Choose or develop AI models that best fit the problem
- **Data Quality**: Ensure the data used for training and operation is clean, relevant, and representative
- **Integration**: Seamlessly integrate the AI component with existing systems and workflows
- **Infrastructure and Tools**: Utilize scalable infrastructure and development tools
- **Testing and Validation**: Rigorously test to ensure it performs as expected under different conditions and scenarios.
- **Ethics and Bias**: Incorporate mechanisms to detect, reduce, and address potential biases
- **Security**: Implement robust security measures to protect the AI app, data, and user information
- **Maintenance and Updates**: Regularly update the AI app to improve functionality, adapt to new data
- **Legal and Compliance**: Complies with relevant legal regulations, industry standards and data privacy policies

Supercharge your AI models

Here are three techniques to use domain expertise and enhance the AI model behind your app

Prompt engineering

In-context learning

Retrieval Augmented Generation

Learn new facts (temporarily)

Fine tuning

Learn new skills (permanently)

Insight.

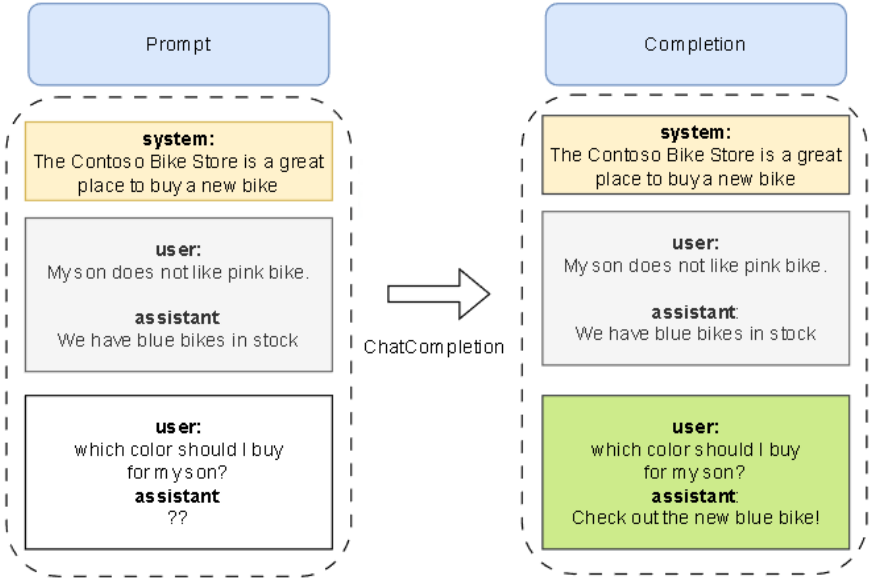
Prompt Engineering

Prompt Payload Structure:

- system prompt
- user prompt
- Assistant

NLP techniques:

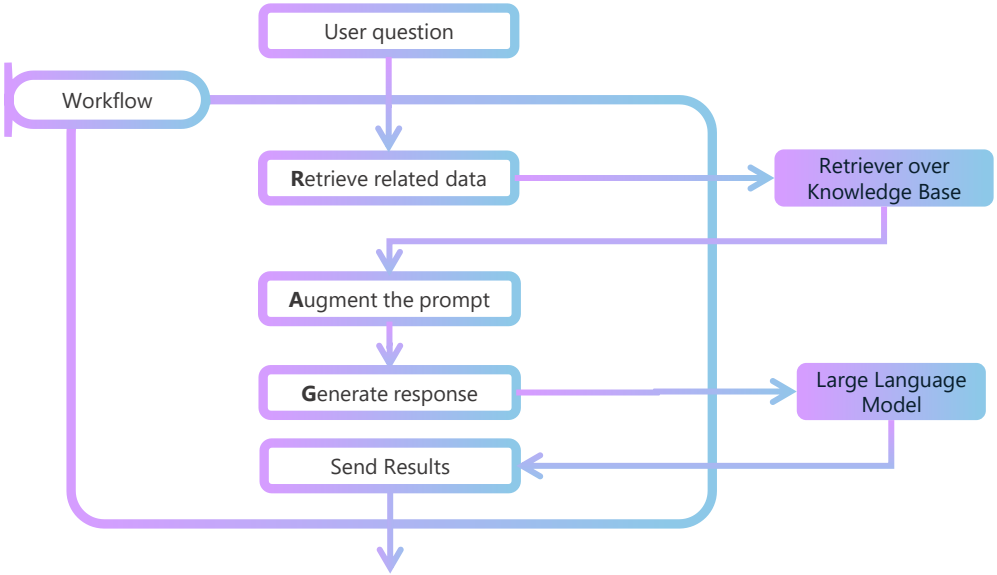
- Zero-shot learning: Provide prompts to an LLM without any prior examples.
- Few-shot learning: Provide prompts to an LLM with a few specific examples of the task.
- Knowledge base in system prompt: Incorporate a knowledge base directly into the system prompt.



Retrieval Augmented Generation

RAG combines generative models with retrieval mechanisms to improve the quality and relevance of generated responses.

- **Retrieval Stage** : Relevant documents or pieces of information are retrieved from a knowledge base or database using a retriever model
- **Generation Stage** : The retrieved documents are used as context for the generative model. The generative model produces a response based on both the input query and the retrieved documents.



Azure AI Search has built-in data chunking and vectorization capabilities

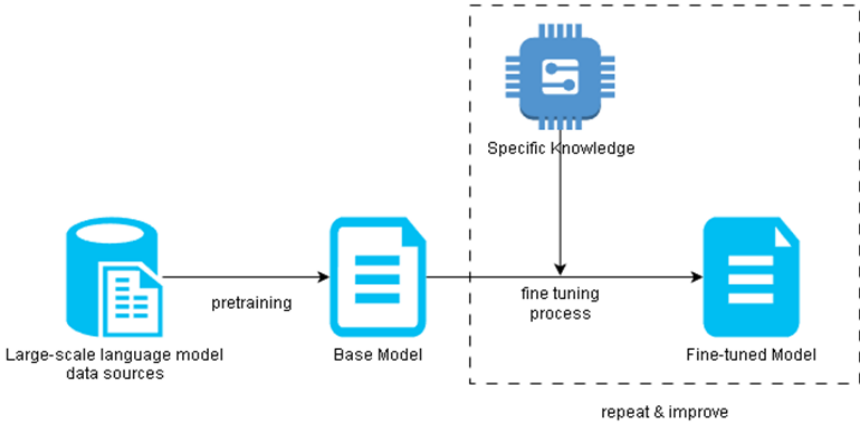
Fine tuning & Custom Model

What's LLM fine-tuning

- Take an open-source or proprietary model
- Re-train it on a variety of concrete examples
- And save the updated weights as a new model checkpoint
- So as to adapt the pre-trained model to specific tasks

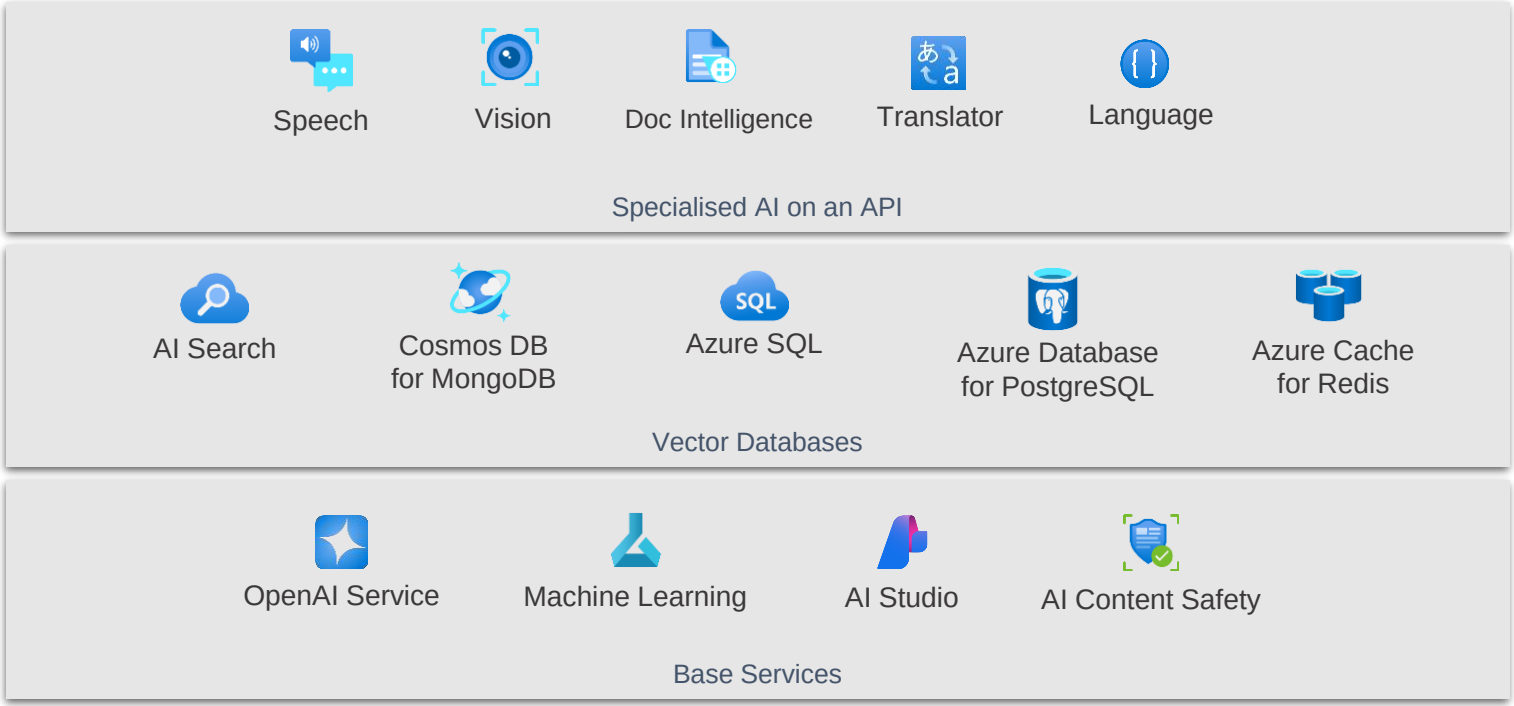
Benefits from fine tuning:

- Higher quality results than prompt engineering
- Train on more examples than can fit into context limit
- Token savings due to shorter prompts
- Lower-latency requests



Azure OpenAI Service supports fine-tuning.

Azure AI services - using AI for Intelligent Apps



Azure OpenAI Service

Industry-Leading Coding and Language AI Models, plus Powerful and Customizable AI Solutions

Many AI Models Available

- GPT-4o: Advanced language model for diverse tasks
- Turbo with Vision: Enhanced model with image understanding capabilities
- Embeddings: For semantic search and context-aware tasks

Access Options

- REST API: Integrate with your existing applications
- Web-Based Interface: User-friendly access for quick tasks

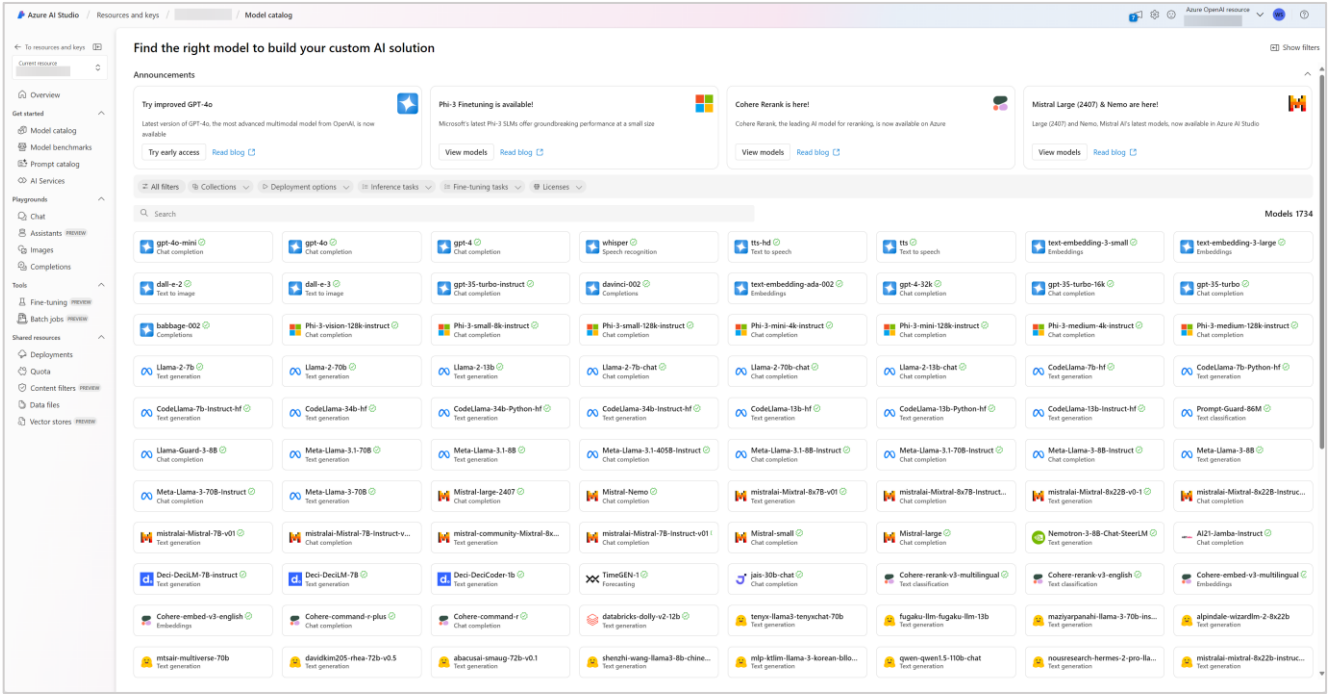
Use Cases

- Content Generation: Create high-quality, relevant content tailored to your needs
- Summarization: Condense information effectively
- Image Understanding: Analyze and interpret visual data
- Semantic Search: Improve search accuracy and relevance

Azure OpenAI Service – not just GPT

Comprehensive Model Library

- Over **2,000** pre-trained models
- Includes state-of-the-art models for diverse applications
- Certain models can be fine-tuned using Azure infrastructure



Azure AI Search (Azure Cognitive Search)

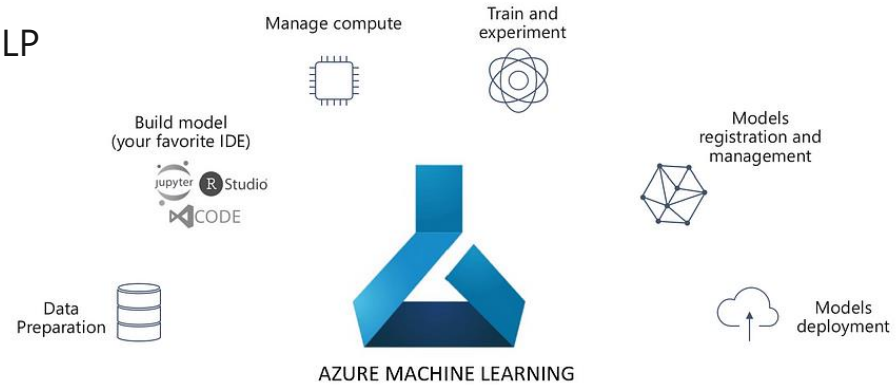
Provides secure information retrieval at scale over user-owned content in traditional and generative AI search applications.

- A search engine for vector search and full text and hybrid search over a search index
- Rich indexing with integrated data chunking and vectorization, lexical analysis for text, and optional applied AI for content extraction and transformation
- Rich query syntax for vector queries, text search, hybrid queries, fuzzy search, autocomplete, geo-search and others
- Relevance and query performance tuning with semantic ranking, scoring profiles, quantization for vector queries, and parameters for controlling query behaviors at runtime
- Azure integration at the data layer, machine learning layer, Azure AI services and Azure OpenAI

Azure Machine Learning

Azure Machine Learning is a comprehensive machine learning platform that supports language model fine-tuning and deployment.

- Automated machine learning: Rapidly create accurate ML models for classification, regression, vision, and NLP
- Model catalog: Discover, fine-tune, and deploy foundation models using model catalog.
- Prompt flow: Design, construct, evaluate, and deploy language model workflows
- AI infrastructure: Purpose-built AI infrastructure



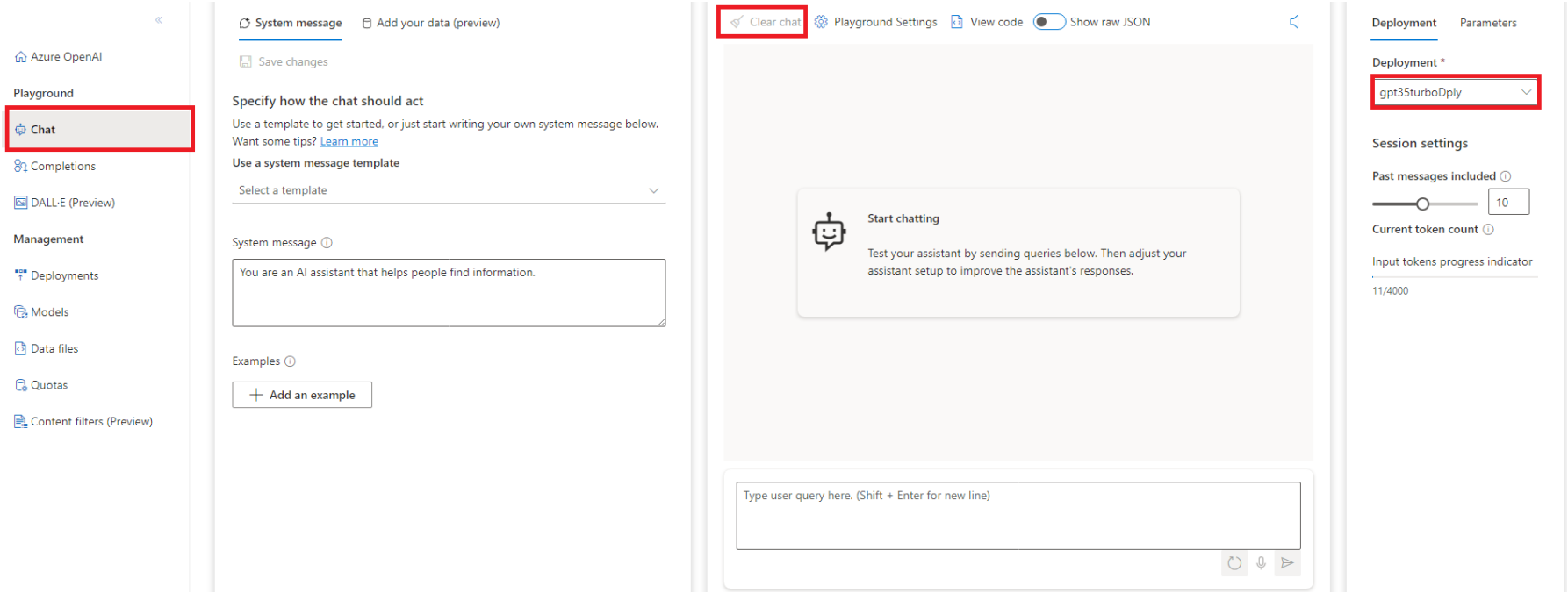
More Azure AI Services

- Azure Content Safety:
  An AI service that detects unwanted contents (text and image)
- Azure Document Intelligence:
  Turn documents into intelligent data-driven solutions.
- Azure Speech / Immersive Reader:
  Speech to text, text to speech, translation, and speaker recognition
- Azure Translator / Language:
  translate more than 100 in-use, at-risk, and endangered languages and dialects.
- Azure Vision & Custom Vision & Face:
  Analyze content in images and videos, Detect and identify people and emotions in images
- Azure Video Indexer:
  Extract actionable insights from your videos
- Azure Bot Service:
  Create bots and connect them across channels.

Azure AI Studio

A unified platform for developing and deploying generative AI apps responsibly

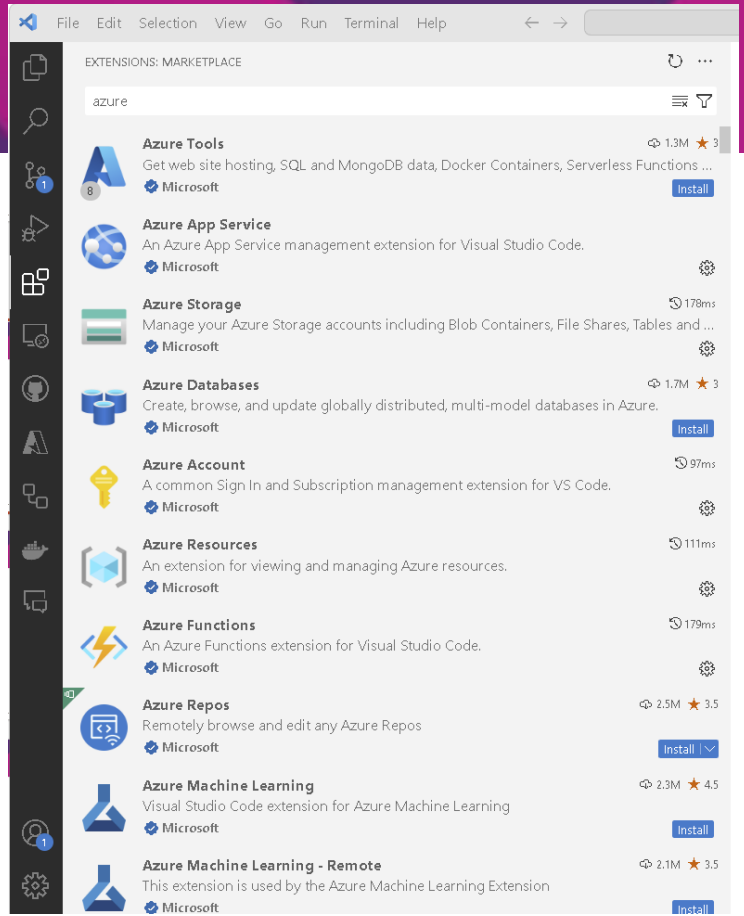
Azure OpenAI Playground



Visual Studio Code

Comprehensive and flexible environment that supports the entire lifecycle of AI app development, from coding and debugging to deployment and maintenance

- Lightweight IDE
- Extensions and Plugins
- Integration with AI and Cloud Services
- Jupyter Notebooks
- GitHub Co-pilot
- Version Control
- Code Snippets and Autocompletion
- Debugging Tools
- Data Visualization
- Customizable Workspace



Azure OpenAI supported languages & libraries

Python
Libraries: openai, requests, http.client

JavaScript/Node.js
Libraries: openai, axios, node-fetch

C# (.NET)
Libraries: Azure.AI.OpenAI

Java
Libraries: azure-ai-openai, okhttp

Go
Libraries: openai-go, net/http

https://python.langchain.com/v0.2/docs/integrations/platforms/microsoft/

Open-source models from Hugging Face Hub to Azure Machine Learning

What are we building today ?

Develop a powerful AI chatbot in one day, utilizing your own data, and leveraging Azure AI Services for scalable, enterprise-ready capabilities

- **Azure AI Services**: Leverage Azure Cognitive Services, including Azure OpenAI for natural language understanding and Cosmos DB (MongoDb) for vector search.
- **Bring Your Own Data**: Utilize your proprietary data sources to tailor the chatbot's responses, making it unique and relevant to your specific use case.
- **Retrieval-Augmented Generation (RAG)**: Implement RAG to enhance the chatbot's responses by retrieving relevant information from your data before generating accurate, context-rich answers.
- **LangChain**: Seamlessly integrate with large language models (LLMs), enabling advanced conversational capabilities and chaining together complex tasks.

Be ambitious.

Tech check

Tech check

Development Environment

- Laptop or Windows Sandbox
- Internet / Wifi
- GitHub repo access
- Visual Studio Code & Extensions
- Node.js / Azure CLI / Bicep

Connections

- Azure Subscription
- Azure OpenAI Subscription Key (shared)
- Cosmos DB connection string (shared)

https://github-insight-anz-lab.github.io/aiapp1day

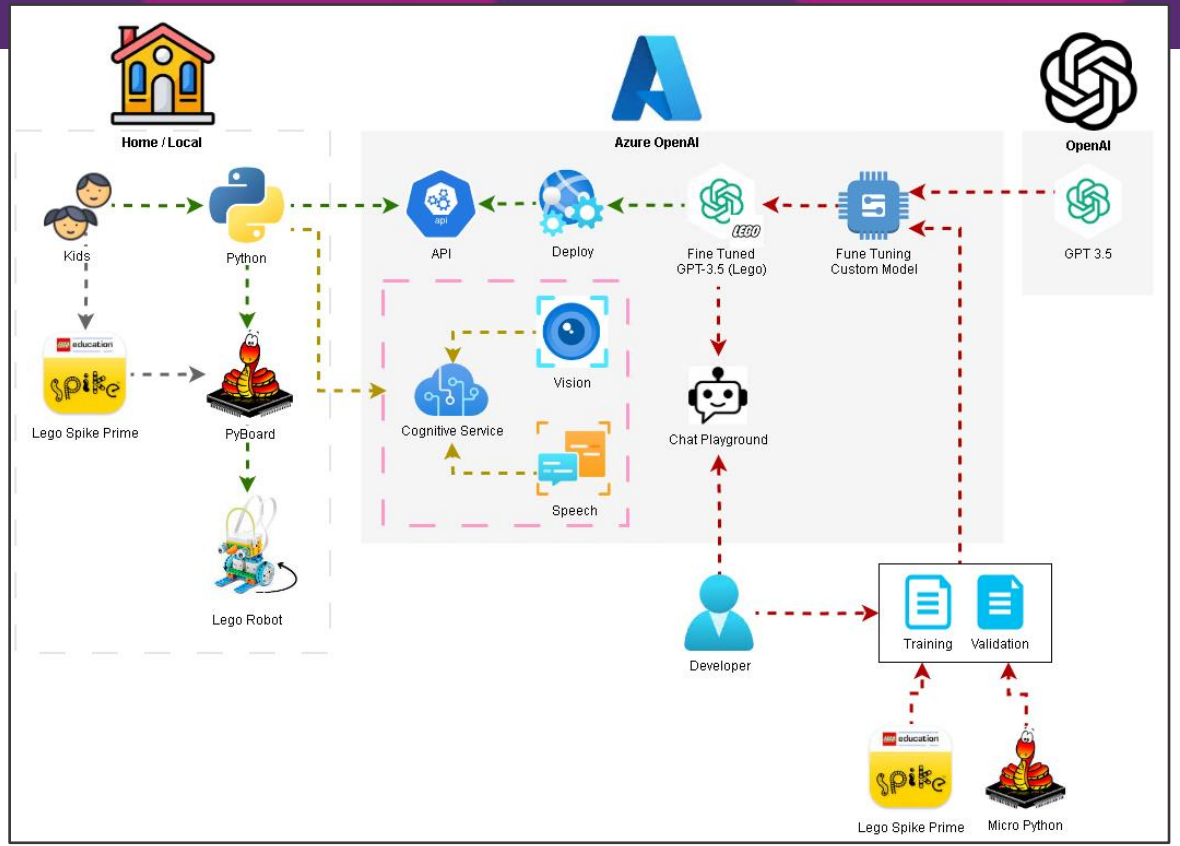
Docusaurus

Share Idea and Experience

Share Idea and Experience

https://devpost.com/software/lego-robot-ai

Share Idea and Experience



Schedule

Time	Activity
08:30 (30m)	Presentation 1: AI Apps using Azure
09:00 (30m)	Tech check
09:30 (30m)	Morning break ☕
10:00 (2h)	Hands-on lab time (Lab 1)
12:00 (30m)	Lunch 🍕 🍔 🥗
12:30 (30m)	Presentation 2: RAG Pattern & Techniques
13:00 (15m)	Lab questions check-in
13:15 (2h)	Hands-on lab time (Lab 2)
15:15 (15m)	Day review and wrap

You are here

Insight.

Be ambitious.

Hands-on lab time - Lab 1

Morning

Lab 1: Interact with Azure OpenAI models

Time: 2 hours

Topics :

- Basic Prompting
- Prompt Engineering Techniques
- System Message
- Add External Knowledge
- Function Calling
- Create Images
- Using Azure OpenAI SDK
- Coding Challenge

YOU WILL NEED

OpenAI Key

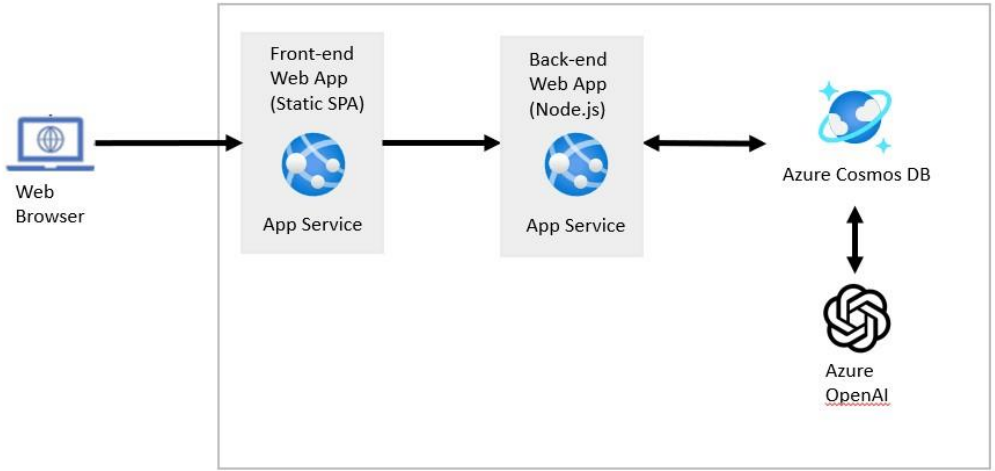
https://github-insight-anz-lab.github.io/aiapp1day

Docusaurus

Lab 2 (Prep) : Azure Deployment

There are 3 options :

- If you have Azure Subscription (Owner or Contributor), you can deploy all resources and use them.
- If you only have limited permission in Azure, you can deploy only app services for the chatbot and using our shared lab resources.
- If you don't have Azure Subscription, you can use our shared lab resources.



Source: https://learn.microsoft.com/dotnet/ai/get-started-app-chat-template

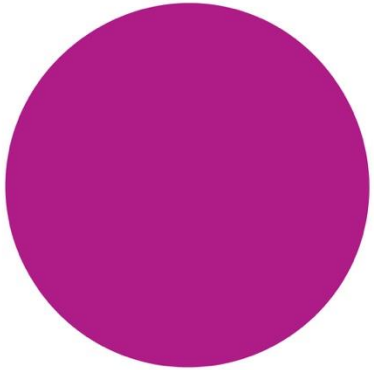
Schedule

Time	Activity
08:30 (30m)	Presentation 1: AI Apps using Azure
09:00 (30m)	Tech check
09:30 (30m)	Morning break ☕
10:00 (2h)	Hands-on lab time (Lab 1)
12:00 (30m)	Lunch 🍕 🍔 🥙
12:30 (30m)	Presentation 2: RAG Pattern & Techniques
13:00 (15m)	Lab questions check-in
13:15 (2h)	Hands-on lab time (Lab 2)
15:15 (15m)	Day review and wrap

You are here

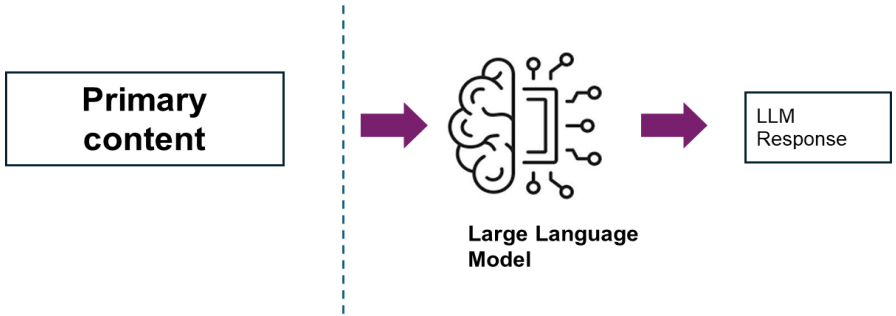
Insight.

Be ambitious.



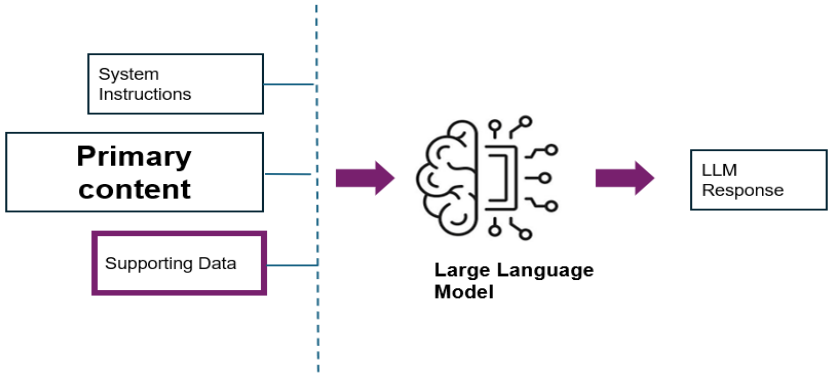
RAG Pattern & Techniques

Limitations of LLM Models



- **Indeterministic** : they can generate different outputs for the same input. This can lead to inconsistency and unpredictability in the results.
- **Hallucination**: LLM models can generate texts that are not based on facts or logic, but on their own learned biases and assumptions. This can lead to false or misleading information in the outputs.
- **Context :** LLM models are trained on public data and are not updated with the latest data and events. They may also lack the specific context or domain knowledge that your application needs.

Retrieval Augmented Generation – Deep Dive

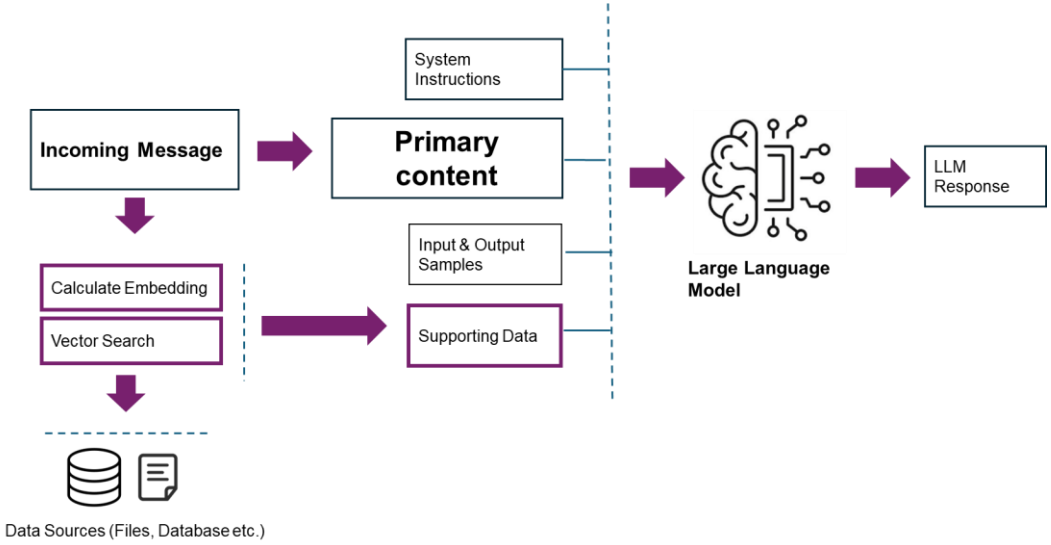


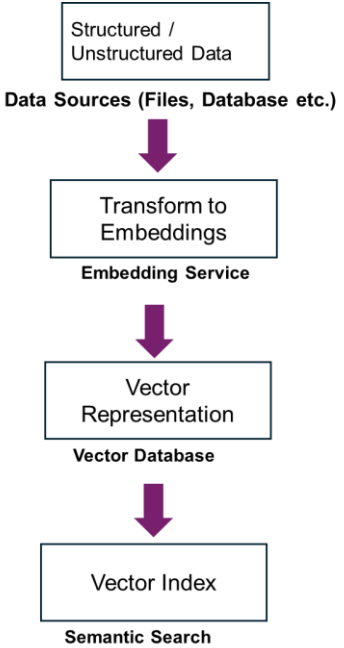

```
You are an AI assistant for a bike store to
help the customer with questions on the products.
The answers should be based on the information provided about the products below -

{{products}}

{{injected_prompt}}
```

RAG combines generative models with retrieval mechanisms to improve the quality and relevance of generated responses.

- **Retrieval Stage** : Relevant documents or pieces of information are retrieved from a knowledge base or database.
- **Generation Stage** : The retrieved documents are used as context for the generative model. The generative model produces a response based on both the input query and the retrieved documents.

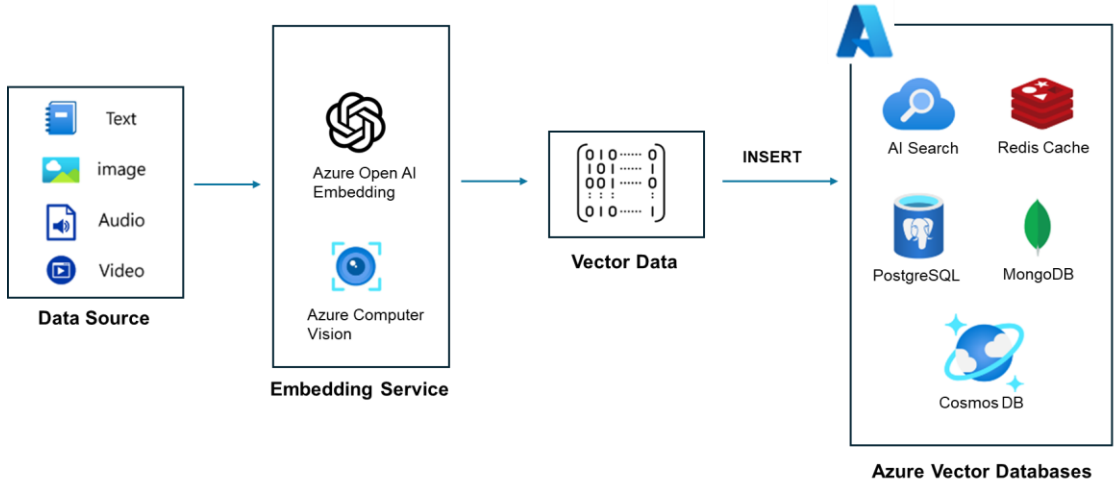
Retrieval Augmented Generation Contd.



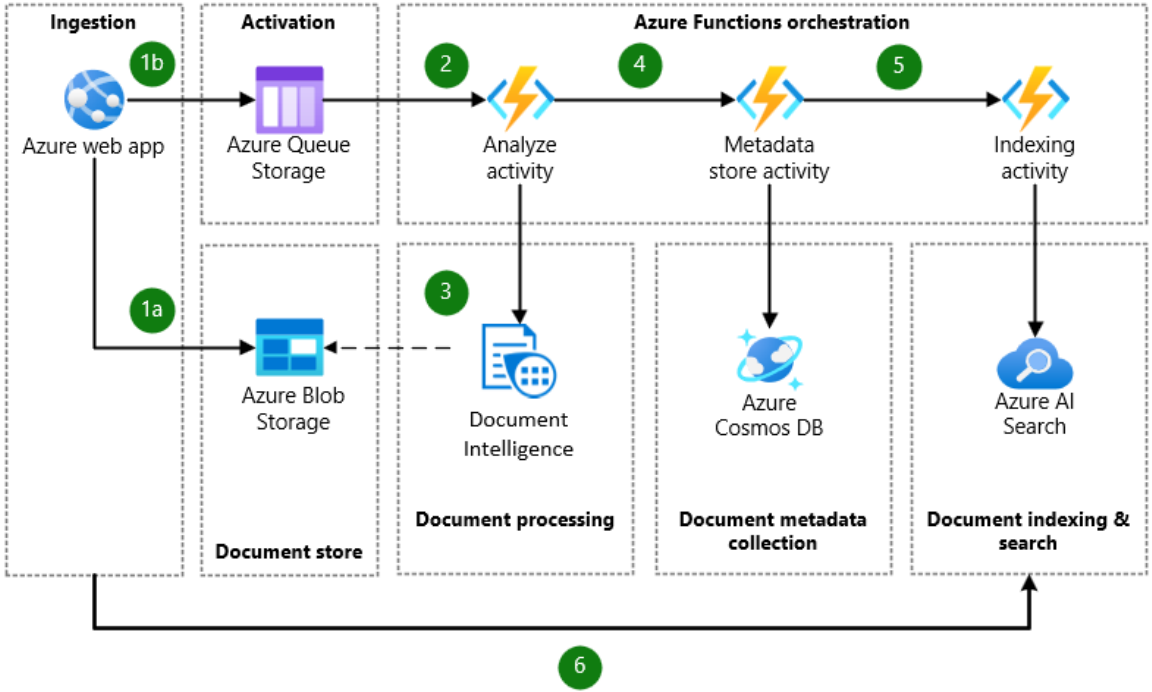


Vectorize text and images in Azure

- Azure offers support for multiple vector databases, including Azure Cache for Redis, Azure AI Search, PostgreSQL, Azure Cosmos, and MongoDB, providing efficient storage and querying capabilities.
- **Embedding Service** transforms text data into vector format.
- **Computer Vision Service** encodes the content and context of an image into vector format.



Document identification and classification

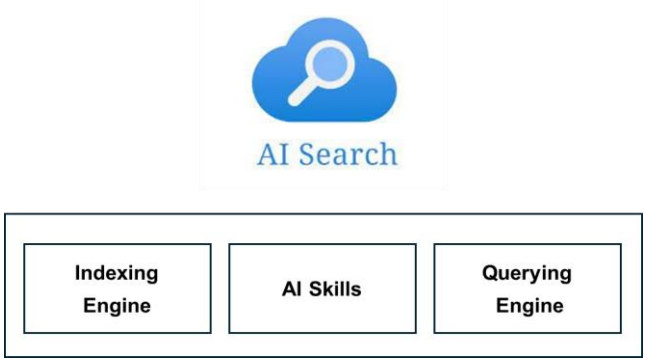


Source: https://learn.microsoft.com/azure/architecture/ai-ml/architecture/automate-document-classification-durable-functions

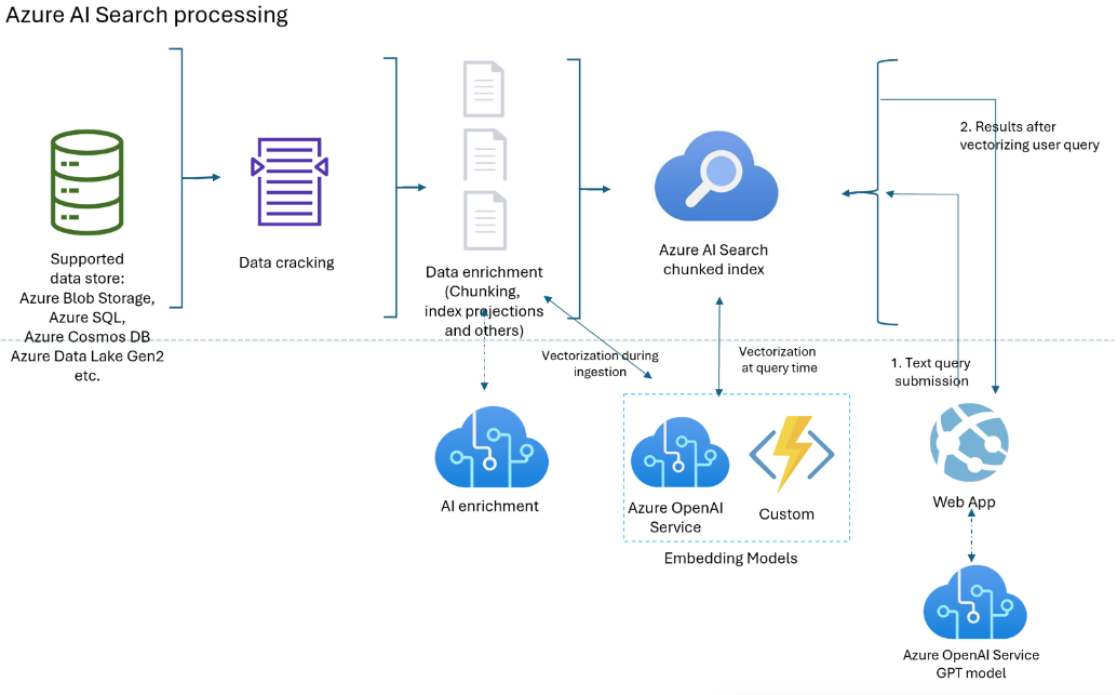
Azure AI Search (Azure Cognitive Search)

Azure AI Search provides secure information retrieval at scale over user-owned content in traditional and generative AI search applications.

- Consolidate heterogeneous content into a search index composed of vectors and text.
- Supports full text, vector and hybrid search over a search index.
- Relevance and query performance tuning with semantic ranking, scoring profiles
- Integrate data chunking and vectorization for generative AI and RAG apps.
- Indexing through AI skills – Transform large unstructured text or image files into searchable chunks
- Apply granular access control at the document level.
- Enterprise-ready - scalability, security and compliance



Azure AI Search – Integrated Vectorization



Integrated vectorization is an extension of the indexing and query pipelines in Azure AI Search.

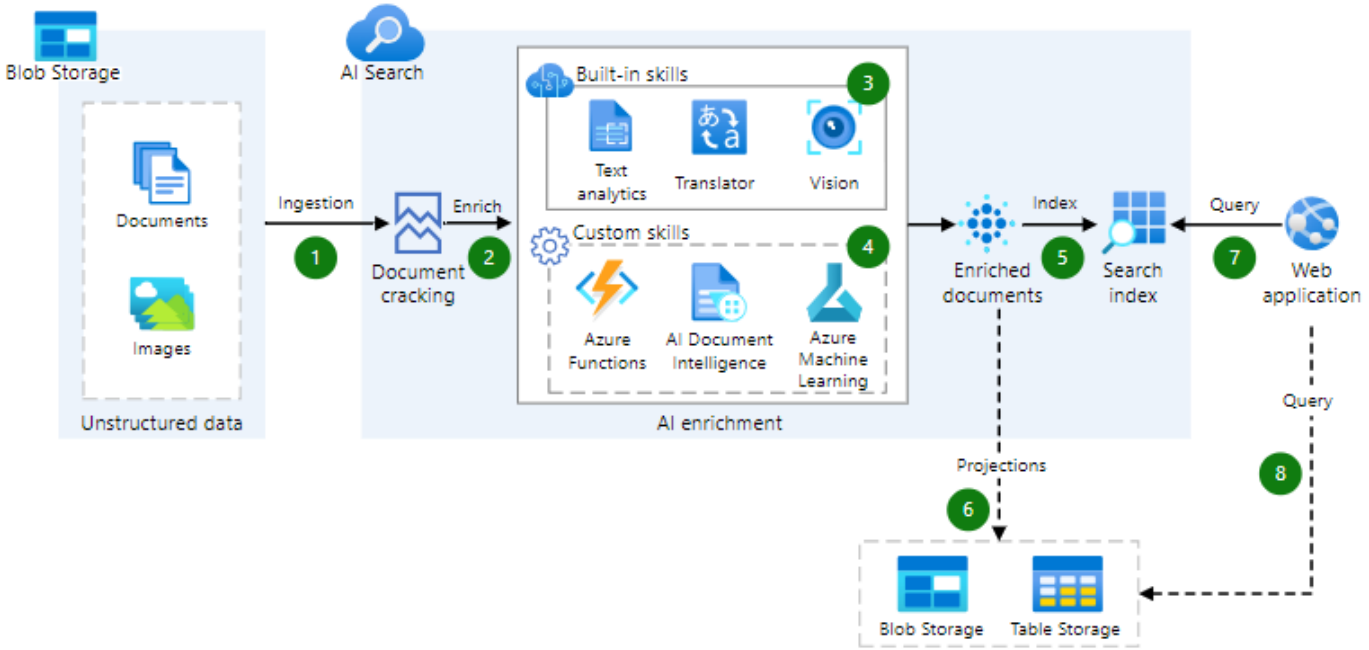
- Data chunking during indexing
- Text-to-vector conversion during indexing
- Text-to-vector conversion during queries

Azure AI Search – AI Skills

A set of skills that prepare a document for indexing, calling either built-in AI search functions or custom code.

- **Entity Recognition:** Identifies and extracts entities such as names, dates, and locations from text.
- **Text Translation:** Translates text from one language to another
- **Text Analytics:** Analyzes text to extract key phrases, sentiments, and language information.
- **OCR (Optical Character Recognition):** Converts images of text into machine-readable text.
- **Image Analysis:** Analyzes images to extract information such as objects, faces, and text contained within them.
- **Sentiment Analysis:** Evaluates text to determine the sentiment or emotional tone, such as positive, neutral, or negative.
- **Language Detection:** Identifies the language of the provided text.
- **PII (Personally Identifiable Information) Detection:** Identifies and masks sensitive information like credit card numbers and social security numbers.
- **Form Recognizer:** Extracts text, key-value pairs, and tables from documents, such as forms and receipts.
- **Custom Skill:** Allows users to define and implement their own custom processing logic.

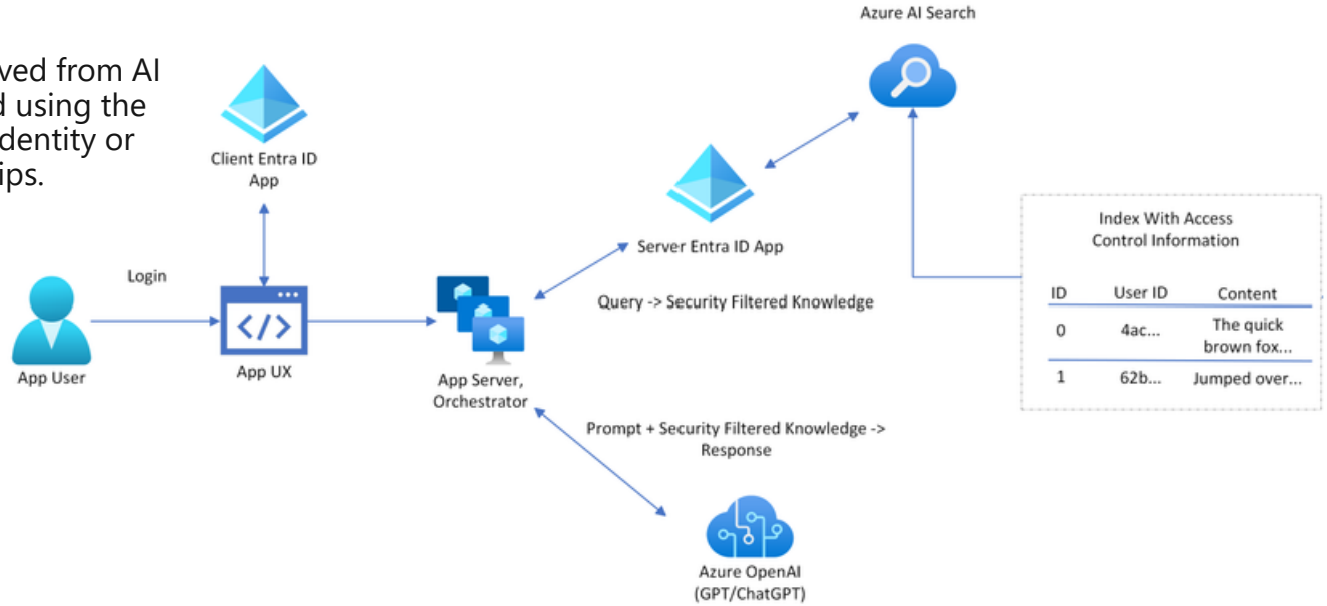
Azure AI Search – Image & Text Processing



https://learn.microsoft.com/en-us/azure/architecture/solution-ideas/articles/ai-search-skillsets

Azure AI Search – Data Security

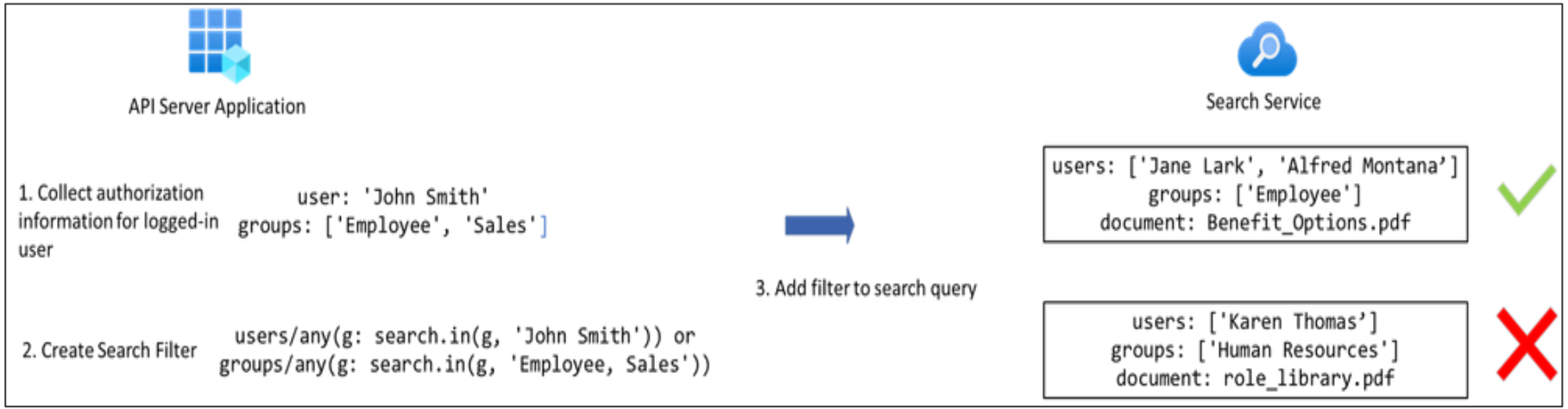
Documents retrieved from AI Search are filtered using the logged-in user's identity or group memberships.



https://techcommunity.microsoft.com/t5/ai-azure-ai-services-blog/access-control-in-generative-ai-applications-with-azure-ai/ba-p/3956408

Azure AI Search – Data Security Contd.

1. The user's identity is extracted from the token claims.
2. The API Server applies a filter containing the authorization information to the query sent to AI Search.



Cosmos DB or Azure AI Search as Vector Database

Azure Cosmos DB is preferred when -

- you have structured or semi-structured data operational data (chat history, customer profile) in that database.
- simplified architecture with vector similarity search inline with database queries.

Azure AI Search is preferred when -

- you need to index structured/unstructured (e.g. images, docx, PDF) from a variety of internal and external data sources
- high quality search results with hybrid full-text / vector search semantic ranking etc.
- the workload requires multi-modal embeddings to perform OCR, image analysis and translation.
- you are building **Bing** like search experience in a custom application

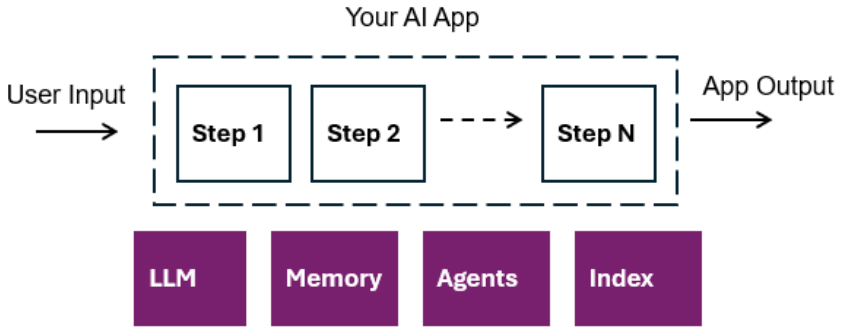
Workflow Orchestration in AI Apps

I like to place an order for the Trek bicycle.
Can you confirm the availability and the price?
And what is the delivery time?

- Find the product in product catalog that matches the user query.
- Check the availability and price of the product.
- Check the delivery time for the product based on the location of the user.

LangChain

Semantic Kernel



Schedule

Time	Activity
08:30 (30m)	Presentation 1: AI Apps using Azure
09:00 (30m)	Tech check
09:30 (30m)	Morning break ☕
10:00 (2h)	Hands-on lab time (Lab 1)
12:00 (30m)	Lunch 🍕 🍔 🥗
12:30 (30m)	Presentation 2: RAG Pattern & Techniques
13:00 (15m)	Lab questions check-in
13:15 (2h)	Hands-on lab time (Lab 2)
15:15 (15m)	Day review and wrap

You are here

Insight.

Be ambitious.

Hands-on lab time – Lab 2

Afternoon

Lab 2: Building a Chatbot using RAG

Time: **2 hours**

Topics :

- Azure Deployment
- Load Product Catalog
- Vector search using text embeddings
- Workflow Orchestration using LangChain
- Build a Chatbot Backend
- Build a Chatbot Frontend
- App Deployment

YOU WILL NEED

OpenAI Key
OpenAI Endpoint
Cosmos DB Login

https://github-insight-anz-lab.github.io/aiapp1day

Docusaurus

Be ambitious.

Wrap up

AI App using Azure

Create AI App power by Azure AI services

- Accelerate app creation with wide range of AI offerings
- Level up your app game and improve your process

Deploy High-Quality AI Models as APIs

- Integrate industry-leading AI models into apps effortlessly
- Enhance user experiences with powerful Generative models

Build and Train Models

- Rapid development with preferred tools and frameworks
- Wide range of models can be trained and fine-tuned easily

Get Your AI App Live

- Leverage Azure’s scalable and reliable services and infrastructure
- Automate processes with Azure services, and get time back in your day

LLM Ops for AI App

Idea Generation

- Define the problem
- Develop a prototype

Technique Selection

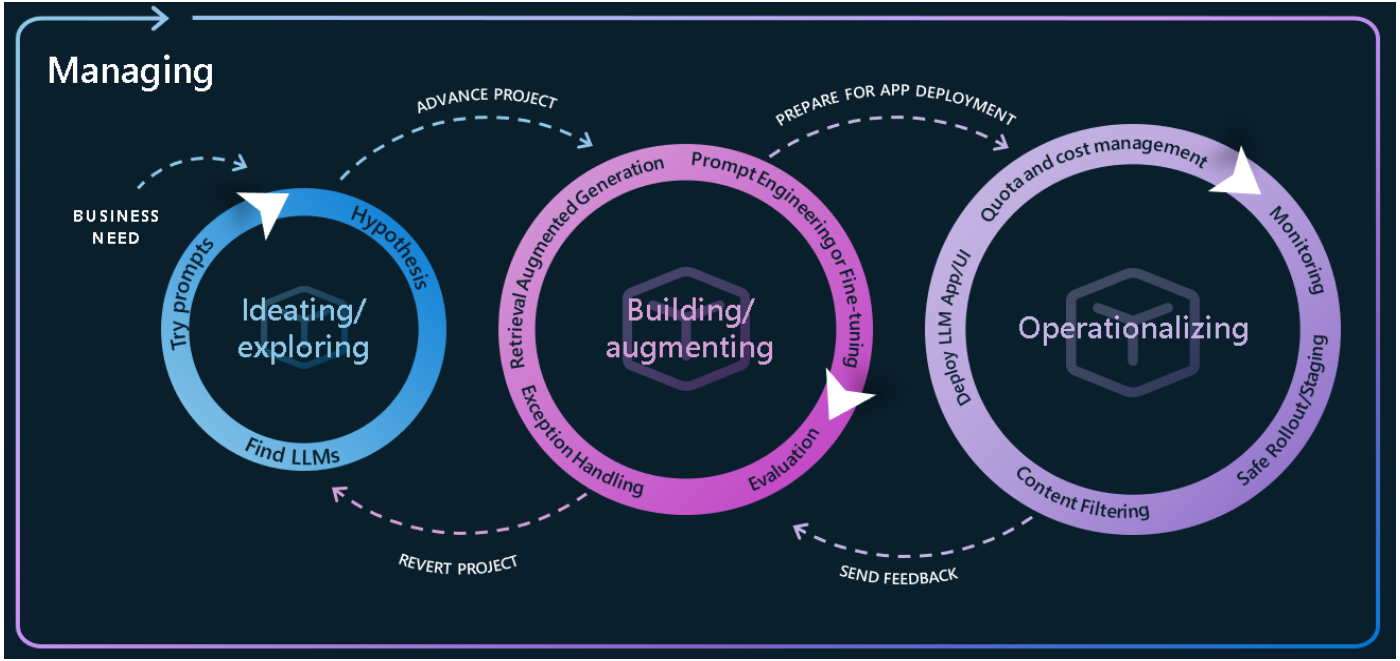
- Model selection
- Dataset Preparation
- Evaluation Metrics

Operationalization

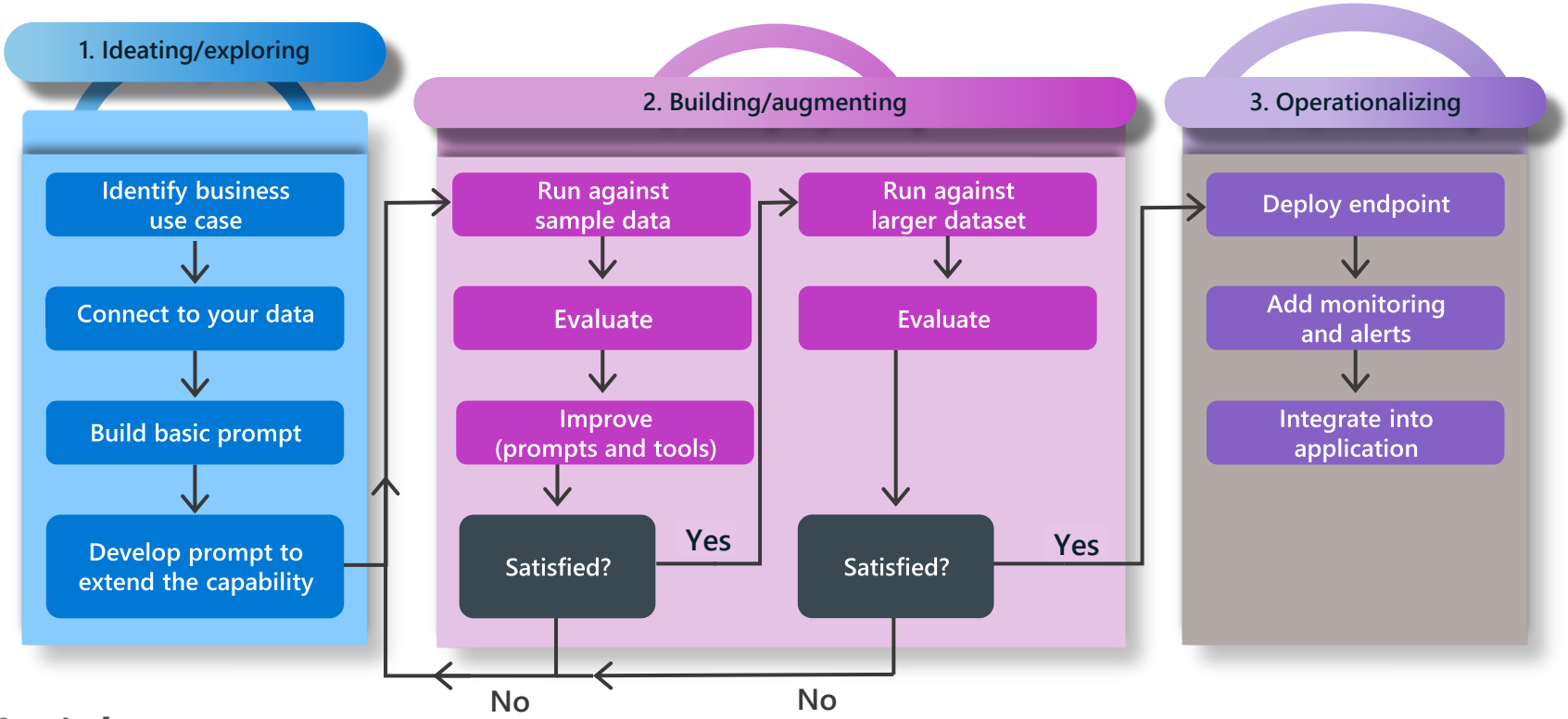
- Integration & Deployment
- Monitoring & Maintenance

Continuous Improvement

- Feedback Loop
- Model Updates
- Adaptation & Optimization



LLM Ops for AI App



Empowering Developers

AI enhances productivity

AI automates repetitive tasks, allowing us to focus on creative problem-solving and high-level design.

Augmenting Human Efforts

AI enhances productivity by handling time-consuming tasks, allowing you to focus on more complex and innovative aspects of the work.

Human expertise is irreplaceable

AI lacks the ability to understand business context, creativity, and nuanced decision-making.

Collaboration between AI and developers:

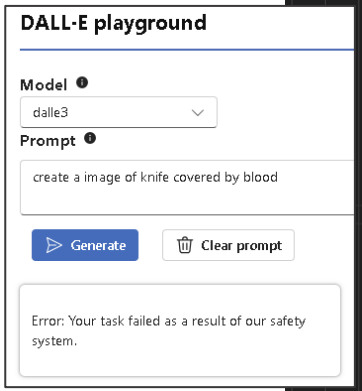
Assisting with coding, debugging, and testing, but developers remain in control of the final product.

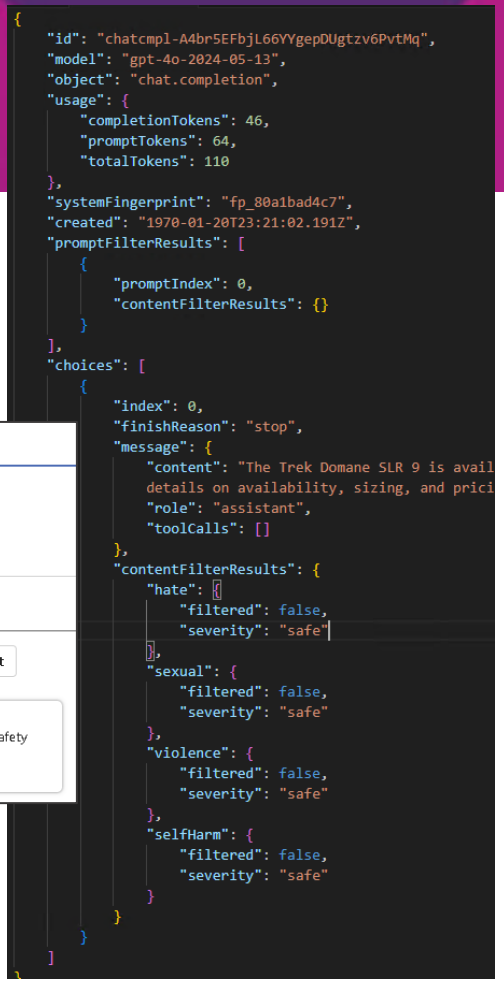
AI + Developers = A Powerful Team

Responsible AI in action

Azure OpenAI Service has a layered approach for generative models, guided by Microsoft's responsible AI principles.

- Integrated safety system provides protection from undesirable inputs and outputs and monitors for misuse
- Provides guidance and best practices for customers to responsibly build applications using these models
- With GPT-4, new research advances from OpenAI have enabled an additional layer of protection
- Guided by human feedback, safety is built directly into the GPT-4 model, which enables the model to be more effective at handling harmful inputs, thereby reducing the likelihood that the model will generate a harmful response





Secure, Compliant, and Trustworthy AI on Azure

Data Security on OpenAI Service with Azure

- **Comprehensive Data Protection**: Azure AI services prioritize data security, ensuring that your information remains confidential and protected.
- **Controlled Access**: Employs a range of security measures, such as encryption in transit and at rest, to safeguard data against threats.
- **Enterprise-Grade Security**: provides industry-leading security features ensuring that your data remains safe throughout processing and storage.
- **Data Privacy & Control**: Does not train models on your data, meaning your business information stays private.
- **Compliance with Global Standards**
- **Identity and Access Management**

Important

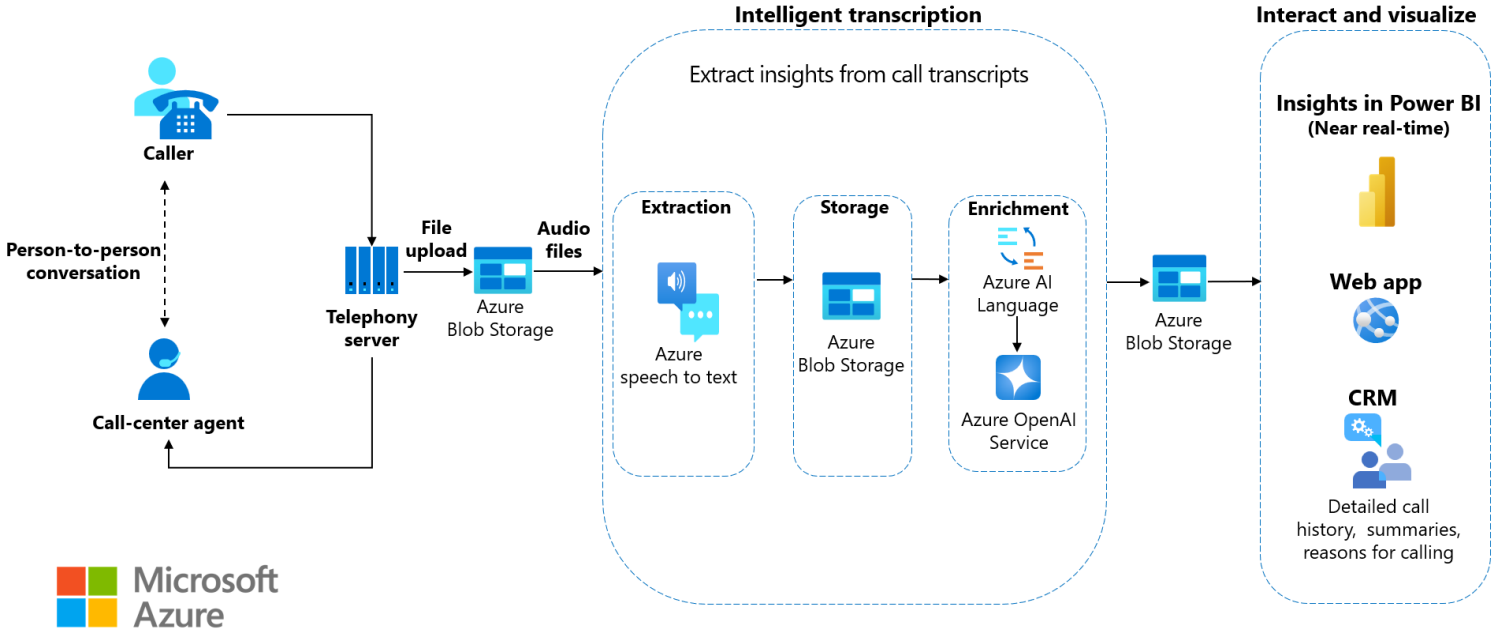
Your prompts (inputs) and completions (outputs), your embeddings, and your training data:

- are NOT available to other customers.
- are NOT available to OpenAI.
- are NOT used to improve OpenAI models.
- are NOT used to train, retrain, or improve Azure OpenAI Service foundation models.
- are NOT used to improve any Microsoft or 3rd party products or services without your permission or instruction.
- Your fine-tuned Azure OpenAI models are available exclusively for your use.

The Azure OpenAI Service is operated by Microsoft as an Azure service; Microsoft hosts the OpenAI models in Microsoft's Azure environment and the Service does NOT interact with any services operated by OpenAI (e.g. ChatGPT, or the OpenAI API).

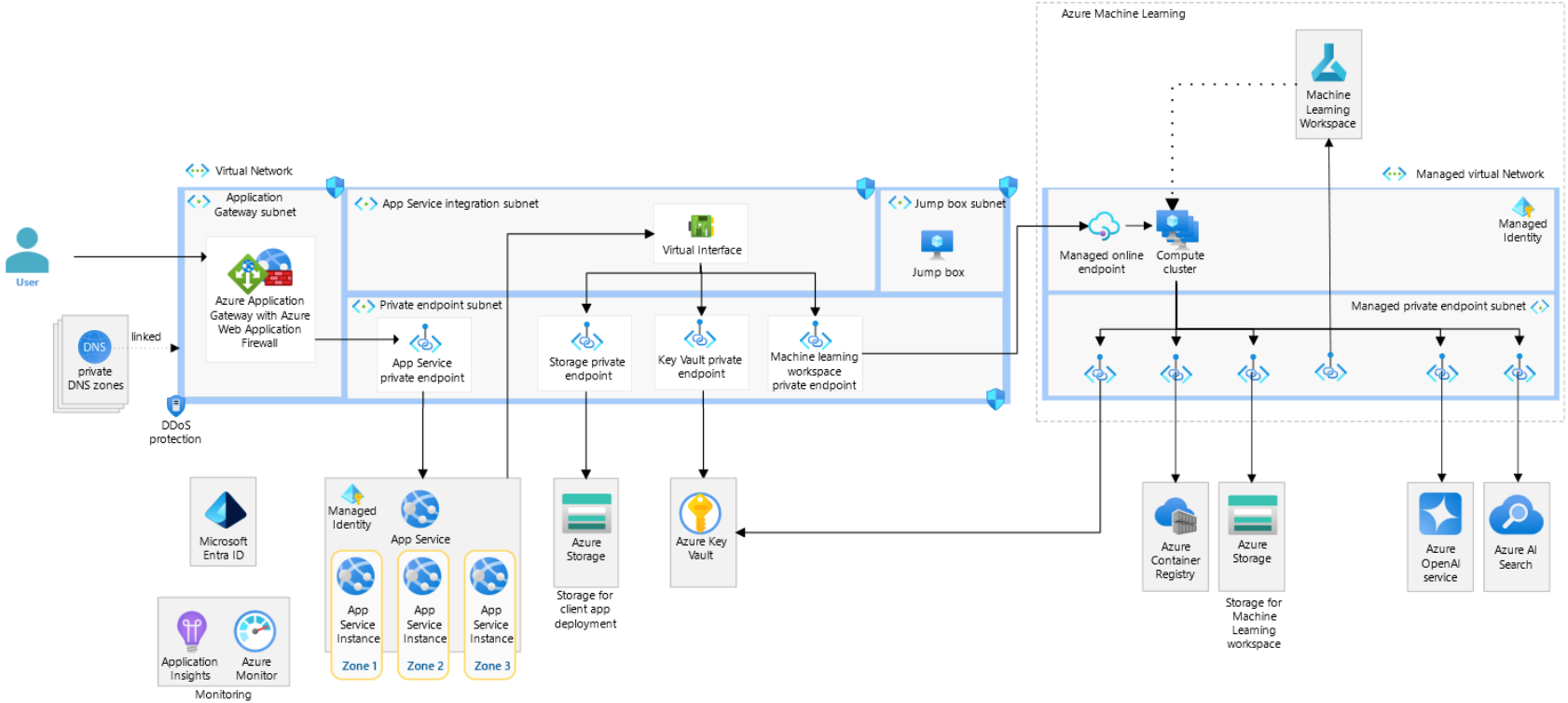
https://learn.microsoft.com/en-us/legal/cognitive-services/openai/data-privacy

Call centre transcript analysis



Source: https://learn.microsoft.com/azure/architecture/ai-ml/openai/architecture/call-center-openai-analytics

Enterprise chat using your data



Source: https://learn.microsoft.com/azure/architecture/ai-ml/architecture/baseline-openai-e2e-chat

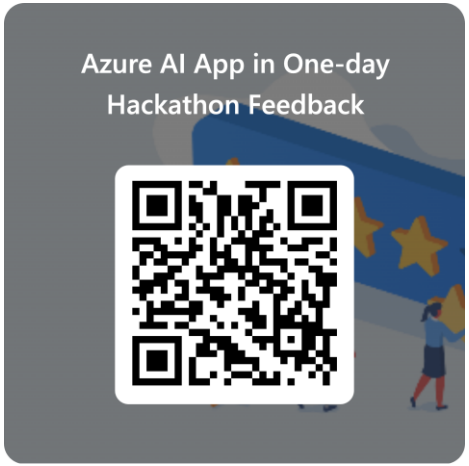
Hackathon in review

Presentations:

- AI App on Azure & Techniques
- Azure’s broad set of AI services
- Supporting capabilities and tools
- RAG Pattern & Techniques
- AI App solutions design

Hands-on Labs:

- Lab 1: Interact with OpenAI Models
- Lab 2: Building a Chatbot using RAG



https://forms.office.com/r/uBEduH1jrd

Next steps

Sign-up Azure Subscription

- Utilize Free Azure Credits

Microsoft Learn

- Engage in AI learning paths and modules
- Build foundational knowledge in AI and Azure services

Hands-on Exercise

- Sample repos on GitHub
- Simple POC

Microsoft Certification

- Microsoft Certified: Azure AI Fundamentals (AI-900)
- Microsoft Certified: Azure AI Engineer Associate (AI-102)

References

The lab & tutorials are modified based on below Microsoft repos:

- Azure OpenAI Cosmos DB hackathon
  - https://github.com/AzureCosmosDB/Azure-OpenAI-Node.js-Developer-Guide
  - https://github.com/AzureCosmosDB/Azure-OpenAI-Developer-Guide-Front-End
- Azure OpenAI Proxy
  - https://github.com/microsoft/Workshop-Interact-with-OpenAI-models/

Other references:

- https://learn.microsoft.com/en-us/azure/architecture/ai-ml/architecture/baseline-openai-e2e-chat
- https://azure.github.io/responsible-ai-hub/docs/content-safety-overview

Be ambitious.

Thank you