Plan and prepare to develop AI solutions on Azure

Learning objectives

Microsoft Azure offers multiple services that enable developers to build amazing AI-powered solutions. Proper planning and preparation involves identifying the services you'll use and creating an optimal working environment for your development team.

By the end of this module, you'll be able to:

- Identify common AI capabilities that you can implement in applications
- Describe **Azure AI Services and considerations** for using them
- Describe **Azure AI Foundry and considerations** for using it
- Identify appropriate **developer tools and SDKs** for an AI project
- Describe considerations for **responsible AI**

Introduction

The growth in the use of artificial intelligence (AI) in general, and generative AI in particular means that developers are increasingly required to create comprehensive AI solutions. These solutions need to combine **machine learning models, AI services, prompt engineering solutions, and custom code**.

Microsoft Azure provides multiple services that you can use to create AI solutions. However, before embarking on an AI application development project, it's useful to consider the available options for services, tools, and frameworks as well as some principles and practices that can help you succeed.

This module explores some of the key considerations for planning an AI development project, and introduces **Azure AI Foundry**; a comprehensive platform for AI development on Microsoft Azure.

What is AI?

The term **"Artificial Intelligence" (AI)** covers a wide range of software capabilities that enable applications to exhibit human-like behavior. AI has been around for many years, and its definition has varied as the technology and use cases associated with it have evolved. In today's technological landscape, AI solutions are built on machine learning models that **encapsulate semantic relationships found in huge quantities of data**; **enabling applications to appear to interpret input in various formats**, **reason over the input data**, and **generate appropriate responses and predictions**.

Common AI capabilities that developers can integrate into a software application include:

Capability	Description
Generative AI	The ability to generate original responses to natural language prompts. For example, software for a real estate business might be used to automatically generate property descriptions and advertising copy for a property listing.
Agents	Generative AI applications that can respond to user input or assess situations autonomously, and **take appropriate actions**. For example, an "executive assistant" agent could provide details about the location of a meeting on your calendar, or even attach a map or automate the booking of a taxi or rideshare service to help you get there.
Computer vision	The ability to accept, interpret, and process visual input from images, videos, and live camera streams. For example, an automated checkout in a grocery store might use computer vision to identify which products a customer has in their shopping basket, eliminating the need to scan a barcode or manually enter the product and quantity.
Speech	The ability to recognize and synthesize speech. For example, a digital assistant might enable users to ask questions or provide audible instructions by speaking into a microphone, and generate spoken output to provide answers or confirmations.
Natural language processing	The ability to process natural language in written or spoken form, analyze it, identify key points, and generate summaries or categorizations. For example, a marketing application might analyze social media messages that mention a particular company, translate them to a specific language, and categorize them as positive or negative based on **sentiment analysis**.

Capability	Description
Information extraction	The ability to use computer vision, speech, and natural language processing to **extract key information from documents, forms, images, recordings, and other kinds of content**. For example, an automated expense claims processing application might extract purchase dates, individual line item details, and total costs from a scanned receipt.
Decision support	The ability to use historic data and learned correlations to make **predictions** that support business decision making. For example, analyzing demographic and economic factors in a city to predict real estate market trends that inform property pricing decisions.

Determining the specific AI capabilities you want to include in your application can help you identify the most appropriate AI services that you'll need to provision, configure, and use in your solution.

A closer look at generative AI

Generative AI represents the latest advance in artificial intelligence, and deserves some extra attention. Generative AI uses language models to respond to natural language prompts, enabling you to build **conversational apps and agents** that support research, content creation, and task automation in ways that were previously unimaginable.

The language models used in generative AI solutions can be **large language models (LLMs)** that have been trained on huge volumes of data and include many millions of parameters; or they can be **small language models (SLMs)** that are optimized for specific scenarios with lower overhead. Language models commonly respond to text-based prompts with natural language text; though increasingly new **multi-modal models** are able to handle image or speech prompts and respond by generating text, code, speech, or images.

Azure AI services

Microsoft Azure provides a wide range of cloud services that you can use to develop, deploy, and manage an AI solution. The most obvious starting point for considering AI development on Azure is **Azure AI services**; a set of out-of-the-box prebuilt APIs and models that you can integrate into your applications. The following table lists some commonly used Azure AI services (for a full list of all available Azure AI services, see Available Azure AI services).

Service	Description
Azure OpenAI	Azure OpenAI in Foundry Models provides access to OpenAI generative AI models including the **GPT family of large and small language models** and **DALL-E image-generation models** within a scalable and securable cloud service on Azure.
Azure AI Vision	The Azure AI Vision service provides a set of models and APIs that you can use to implement common computer vision functionality in an application. With the AI Vision service, you can **detect common objects in images, generate captions, descriptions, and tags based on image contents, and read text in images**.
Azure AI Speech	The Azure AI Speech service provides APIs that you can use to implement **text to speech** and **speech to text transformation**, as well as specialized speech-based capabilities like **speaker recognition and translation**.
Azure AI Language	The Azure AI Language service provides models and APIs that you can use to analyze natural language text and perform tasks such as **entity extraction, sentiment analysis, and summarization**. The AI Language service also provides functionality to help you **build conversational language models and question answering solutions**.
Azure AI Content Safety	Azure AI Content Safety provides developers with access to advanced algorithms for processing images and text and **flagging content that is potentially offensive, risky, or otherwise undesirable**.
Azure AI Translator	The Azure AI Translator service uses state-of-the-art language models to **translate text** between a large number of languages.
Azure AI Face	The Azure AI Face service is a specialist computer vision implementation that can **detect, analyze, and recognize human faces**. Because of the potential risks associated with personal identification and misuse of this capability, access to some features of the AI Face service are restricted to approved customers.
Azure AI Custom Vision	The Azure AI Custom Vision service enables you to train and use custom computer vision models for **image classification and object detection**.
Azure AI Document Intelligence	With Azure AI Document Intelligence, you can use pre-built or custom models to **extract fields from complex documents such as invoices, receipts, and forms**.

Service	Description
Azure AI Content Understanding	The Azure AI Content Understanding service provides **multi-modal content analysis** capabilities that enable you to build models to extract data from forms and documents, images, videos, and audio streams.
Azure AI Search	The Azure AI Search service uses a pipeline of AI skills based on other Azure AI Services and custom code to **extract information from content and create a searchable index**. AI Search is commonly used to **create vector indexes for data** that can then be used to ground prompts submitted to generative AI language models, such as those provided in the Azure OpenAI service.

Considerations for Azure AI services resources

To use Azure AI services, you create one or more Azure AI resources in an Azure subscription and implement code in client applications to consume them. In some cases, AI services include **web-based visual interfaces** that you can use to configure and test your resources - for example to train a custom image classification model using the Custom Vision service you can use the visual interface to upload training images, manage training jobs, and deploy the resulting model.

> Note: You can provision Azure AI services resources in the Azure portal (or by using BICEP or ARM templates or the Azure command-line interface) and build applications that use them directly through **various service-specific APIs and SDKs**. However, as we'll discuss later in this module, in most medium to large-scale development scenarios it's better to provision Azure AI services resources as part of an **Azure AI Foundry project** - enabling you to centralize access control and cost management, and making it easier to manage shared resources and build the next generation of generative AI apps and agents.

Single service or multi-service resource?

Most Azure AI services, such as **Azure AI Vision, Azure AI Language**, and so on, can be provisioned as standalone resources, enabling you to create only the Azure resources you specifically need. Additionally, **standalone Azure AI services often include a free-tier SKU with limited functionality**, enabling you to evaluate and develop with the service at no cost. Each standalone Azure AI resource provides **an endpoint and authorization keys** that you can use to access it securely from a client application.

Alternatively, you can provision a **multi-service resource** that **encapsulates multiple AI services in a single Azure resource**. Using a multi-service resource can make it easier to manage applications that use multiple AI capabilities. There are two multi-service resource types you can use:

1. Azure AI services

The Azure AI Services resource type includes the following services, making them available from a single endpoint:

- Azure AI Speech
- Azure AI Language
- Azure AI Translator
- Azure AI Vision
- Azure AI Face
- Azure AI Custom Vision
- Azure AI Document Intelligence

2. Azure AI Foundry

The Azure AI Foundry resource type includes the following services, and supports working with them through an Azure AI Foundry project:

- Azure OpenAI
- Azure AI Speech
- Azure AI Language
- Azure AI Foundry Content Safety
- Azure AI Translator
- Azure AI Vision
- Azure AI Face
- Azure AI Document Intelligence
- Azure AI Content Understanding

Using a multi-service resource can make it easier to manage applications that use multiple AI capabilities.

Regional availability

Some services and models are available in only a subset of Azure regions. Consider service availability and any regional quota restrictions for your subscription when provisioning Azure AI services. Use the product availability table to check regional availability of Azure services. Use the model availability table in the Azure OpenAI service documentation to determine regional availability for Azure OpenAI models.

Cost

Azure AI services are charged based on usage, with different pricing schemes available depending on the specific services being used. As you plan an AI solution on Azure, use the Azure AI services pricing documentation to understand pricing for the AI services you intend to incorporate into your application. You can use the Azure pricing calculator to estimate the costs your expected usage will incur.

Azure AI Foundry

Azure AI Foundry is a platform for AI development on Microsoft Azure. While you can provision individual Azure AI services resources and build applications that consume them without it, the project organization, resource management, and AI development capabilities of Azure AI Foundry makes it the recommended way to build all but the most simple solutions.

Azure AI Foundry provides the *Azure AI Foundry portal*, a web-based visual interface for working with AI projects. It also provides the *Azure AI Foundry SDK*, which you can use to build AI solutions programmatically.

Azure AI Foundry projects

In Azure AI Foundry, you manage the resource connections, data, code, and other elements of the AI solution in projects. There are two kinds of project:

Foundry projects

Foundry projects are associated with an **Azure AI Foundry** resource in an Azure subscription. Foundry projects provide support for *Azure AI Foundry models (including OpenAI models), Azure AI Foundry Agent Service, Azure AI services, and tools for evaluation and responsible AI development*.

An Azure AI Foundry resource supports the most common AI development tasks to develop generative AI chat apps and agents. In most cases, using a Foundry project provides the right level of resource centralization and capabilities with a minimal amount of administrative resource management. You can use Azure AI Foundry portal to work in projects that are based in Azure AI Foundry resources, making it easy to add connected resources and manage model and agent deployments.

Hub-based projects

Hub-based projects are associated with an **Azure AI hub** resource in an Azure subscription. Hub-based projects include an Azure AI Foundry resource, as well as managed compute, support for

Prompt Flow development, and connected **Azure storage** and **Azure key vault** resources for secure data storage.

Azure AI hub resources support advanced AI development scenarios, like developing **Prompt Flow based applications or fine-tuning models**. You can also **use Azure AI hub resources in both Azure AI Foundry portal and Azure Machine learning portal**, making it easier to work on collaborative projects that involve data scientists and machine learning specialists as well as developers and AI software engineers

> Tip: For more information about Azure AI Foundry project types, see What is Azure AI Foundry?.
> Note: the notes below are from the old course.

Hubs and projects

In Azure AI Foundry, you manage the resources, assets, code, and other elements of the AI solution in hubs and projects. **Hubs provide a top-level container for managing shared resources, data, connections and security configuration for AI application development**. A hub can support multiple projects, in which developers collaborate on building a specific solution.

Hubs

A hub provides a centrally managed collection of shared resources and management configuration for AI solution development. You need at least one hub to use all of the solution development features and capabilities of AI Foundry.

In a hub, you can define shared resources to be used across multiple projects. When you create a hub using the Azure AI Foundry portal, an **Azure AI Hub** resource is created in a resource group associated with the hub. Additionally, the following resources are created for the hub:

- A multi-service **Azure AI services** resource to provide access to Azure OpenAI and other Azure AI services.
- A **Key vault** in which sensitive data such as connections and credentials can be stored securely.
- A **Storage account** for data used in the hub and its projects.
- Optionally, an **Azure AI Search** resource that can be used to index data and support grounding for generative AI prompts.

You can create more resources as required (for example, an **Azure AI Face** resource) and add it to the hub (or an individual project) by defining a connected resource. As you create more items in your hub, such as compute instances or endpoints, more resources will be created for them in the Azure resource group.

Access to the resources in a hub is governed by creating *users* and assigning them to *roles*. An IT administrator can manage access to the resources centrally at the hub level, and projects associated with the hub inherit the resources and role assignments; enabling development teams to use the resources they need without needing to request access on a project-by-project basis.

Projects

A hub can support one or more projects, each of which is used to organize the resources and assets required for a particular AI development effort.

Users can collaborate in a project, sharing data in project-specific storage containers and connected resources, and using the shared resources defined in the hub associated with the project. Azure AI Foundry provides tools and functionality within a project that developers can use to build AI solutions efficiently, including:

- A **model catalog** in which you can find and deploy machine learning models from multiple sources, including Azure OpenAI and the Hugging Face model library.
- **Playgrounds** in which you can test prompts with generative AI models.
- Access to **Azure AI services**, including visual interfaces to experiment with and configure services as well as endpoints and keys that you can use to connect to them from client applications.
- **Visual Studio Code** containers that define a hosted development environment in which you can write, test, and deploy code.
- **Fine-tuning** functionality for generative AI models that you need to customize based on custom training prompts and responses.
- **Prompt Flow**, a prompt orchestration tool that you can use to define the logic for a generative AI application's interaction with a model.
- Tools to assess, evaluate, and improve your AI applications, including *tracing, evaluations, and content safety and security management*.
- Management of project **assets**, including models and endpoints, data and indexes, and deployed web apps.

Considerations for Azure AI Foundry

When planning an AI solution built on Azure AI Foundry, there are some additional considerations to those discussed previously in relation to Azure AI services.

Hub and project organization

Plan your hub and project organization for the most effective management of resources and efficiency of administration. **Use Hubs to centralize management of users and shared resources that are involved in related projects, and then add project-specific resources as necessary.** For example, an organization might have separate software development teams for each area of the business, so it may make sense to create separate hubs for each business area (such as Marketing, HR, and so on) in which AI application development projects for each business area can be created. The shared resources in each hub will automatically be available in projects created in those hubs.

Connected resources

At the hub level, an IT administrator can create shared resource connections in a hub that will be used in downstream projects. Projects access the connected resources by proxy on behalf of project users, so users in those projects don't need direct access to those resources in order to use them within the context of the project. Connections in a hub are automatically available in new projects in the hub without further requests to the IT administrator. If an individual project needs access to a specific resource that other projects in the same hub don't use, you can create more connected resources at the project level.

As you plan your Azure AI Foundry hubs and projects, identify the shared connected resources you should add to each hub so that they're inherited by projects in that hub, while allowing for project-level exceptions.

Security and authorization

For each hub and project, identify the users who will need access and the roles to which they should be assigned.

Hub-level roles

Hub-level roles can perform infrastructure management tasks, such as creating hub-level connected resources or new projects. The default roles in a hub are:

- **Owner**: Full access to the hub, including the ability to manage and create new hubs and assign permissions. This role is automatically assigned to the hub creator
- **Contributor**: Full access to the hub, including the ability to create new hubs, but isn't able to manage hub permissions on the existing resource.
- **Azure AI Developer**: All permissions except create new hubs and manage the hub permissions.
- **Azure AI Inference Deployment Operator**: All permissions required to create a resource deployment within a resource group.

- **Reader**: Read only access to the hub. This role is automatically assigned to all project members within the hub.

Project-level roles

Project-level roles determine the tasks that a user can perform within an individual project. The default roles in a project are:

- **Owner**: Full access to the project, including the ability to assign permissions to project users.
- **Contributor**: Full access to the project but can't assign permissions to project users.
- **Azure AI Developer**: Permissions to perform most actions, including create deployments, but can't assign permissions to project users.
- **Azure AI Inference Deployment Operator**: Permissions to perform all actions required to create a resource deployment within a resource group.
- **Reader**: Read only access to the project.

Regional availability

As with all Azure services, the availability of specific Azure AI Foundry capabilities can vary by region. As you plan your solution, determine regional availability for the capabilities you require.

Costs and quotas

In addition to the cost of the Azure AI services your solution uses, there are costs associated with Azure AI Foundry related to the resources that support hubs and projects as well as storage and compute for assets, development, and deployed solutions. You should consider these costs when planning to use Azure AI Foundry for AI solution development.

In addition to service consumption costs, you should consider the resource quotas you need to support the AI applications you intend to build. Quotas are used to limit utilization, and play a key role in cost management and managing Azure capacity. In some cases, you may need to request additional quota to increase rate limits for AI model operations or available compute for development and solution deployment.

Developer tools and SDKs

While you can perform many of the tasks needed to develop an AI solution directly in the Azure AI Foundry portal, developers also need to write, test, and deploy code.

Development tools and environments

There are many development tools and environments available, and developers should choose one that supports the languages, SDKs, and APIs they need to work with and with which they're most comfortable. For example, a developer who focuses strongly on building applications for Windows using the .NET Framework might prefer to work in an integrated development environment (IDE) like **Microsoft Visual Studio**. Conversely, a web application developer who works with a wide range of open-source languages and libraries might prefer to use a code editor like **Visual Studio Code (VS Code)**. Both of these products are suitable for developing AI applications on Azure.

The Azure AI Foundry for Visual Studio Code extension

When developing Azure AI Foundry based generative AI applications in Visual Studio Code, you can use the **Azure AI Foundry for Visual Studio Code extension** to simplify key tasks in the workflow, including:

- Creating a project.
- Selecting and deploying a model.
- Testing a model in the playground.
- Creating an agent.

> Tip: For more information about using the Azure AI Foundry for Visual Studio Code extension, see Work with the Azure AI Foundry for Visual Studio Code extension.

The Azure AI Foundry VS Code container image

As an alternative to installing and configuring your own development environment, within Azure AI Foundry portal, you can create compute and use it to host a container image for VS Code (installed locally or as a hosted web application in a browser). The benefit of using the container image is that it includes the latest versions of the SDK packages you're most likely to work with when building AI applications with Azure AI Foundry.

GitHub and GitHub Copilot

GitHub is the world's most popular platform for **source control and DevOps management**, and can be a critical element of any team development effort. Visual Studio and VS Code (including the Azure AI Foundry VS Code container image) both provide native integration with GitHub, and access to GitHub Copilot; an AI assistant that can significantly improve developer productivity and effectiveness.

> Tip: For more information about using GitHub Copilot in Visual Studio Code, see GitHub Copilot in VS Code.

Programming languages, APIs, and SDKs

You can develop AI applications using many common programming languages and frameworks, including **Microsoft C#, Python, Node, TypeScript, Java**, and others. When building AI solutions on Azure, some common SDKs you should plan to install and use include:

- The Azure AI Foundry SDK, which enables you to write code to connect to Azure AI Foundry projects and access resource connections, which you can then work with using service-specific SDKs.
- The Azure AI Foundry Models API, which provides an interface for working with generative AI model endpoints hosted in Azure AI Foundry.
- The Azure OpenAI in Azure AI Foundry Models API, which enables you to build chat applications based on OpenAI models hosted in Azure AI Foundry.
- Azure AI Services SDKs - AI service-specific libraries for multiple programming languages and frameworks that enable you to consume Azure AI Services resources in your subscription. You can also use Azure AI Services through their REST APIs.
- The Azure AI Foundry Agent Service, which is accessed through the Azure AI Foundry SDK and can be integrated with frameworks like Semantic Kernel to build comprehensive AI agent solutions.

Responsible AI

It's important for software engineers to consider the impact of their software on users, and society in general; including considerations for its responsible use. When the application is imbued with artificial intelligence, these considerations are particularly important due to the nature of how AI systems work and inform decisions; often based on probabilistic models, which are in turn dependent on the data with which they were trained.

The human-like nature of AI solutions is a significant benefit in making applications user-friendly, but it can also lead users to place a great deal of trust in the application's ability to make correct decisions. **The potential for harm to individuals or groups through incorrect predictions or misuse of AI capabilities is a major concern**, and software engineers building AI-enabled solutions should apply due consideration to mitigate risks and ensure fairness, reliability, and adequate protection from harm or discrimination.

Let's discuss some core principles for responsible AI that have been adopted at Microsoft.

Fairness

AI systems should treat all people fairly. For example, suppose you create a machine learning model to support a loan approval application for a bank. The model should make predictions of whether or not the loan should be approved **without incorporating any bias** based on gender, ethnicity, or other factors that might result in an unfair advantage or disadvantage to specific groups of applicants.

Fairness of machine learned systems is a highly active area of ongoing research, and some software solutions exist for evaluating, quantifying, and mitigating unfairness in machine learned models. However, tooling alone isn't sufficient to ensure fairness. **Consider fairness from the beginning of the application development process; carefully reviewing training data to ensure it's representative of all potentially affected subjects, and evaluating predictive performance for subsections of your user population throughout the development lifecycle**.

Reliability and safety

AI systems should **perform reliably and safely**. For example, consider an AI-based software system for an autonomous vehicle; or a machine learning model that diagnoses patient symptoms and recommends prescriptions. Unreliability in these kinds of system can result in substantial risk to human life.

As with any software, AI-based software application development must be subjected to rigorous testing and deployment management processes to ensure that they **work as expected** before release. Additionally, software engineers need to **take into account the probabilistic nature of machine learning models**, and apply appropriate thresholds when **evaluating confidence scores for predictions**.

Privacy and security

AI systems should be secure and respect privacy. The machine learning models on which AI systems are based rely on large volumes of data, which may contain personal details that must be kept private. Even after models are trained and the system is in production, they use new data to make predictions or take action that may be subject to privacy or security concerns; so **appropriate safeguards to protect data and customer content** must be implemented.

Inclusiveness

AI systems should **empower everyone and engage people**. AI should bring benefits to all parts of society, regardless of physical ability, gender, sexual orientation, ethnicity, or other factors.

One way to optimize for inclusiveness is to ensure that the *design, development, and testing of your application includes input from as diverse a group of people as possible*.

Transparency

AI systems should be understandable. Users should be made fully aware of the purpose of the system, how it works, and what limitations may be expected.

For example, when an AI system is based on a machine learning model, you should generally make users aware of **factors that may affect the accuracy of its predictions**, such as the number of cases used to train the model, or the specific features that have the most influence over its predictions. You should also share information about the **confidence score** for predictions.

When an AI application relies on personal data, such as a facial recognition system that takes images of people to recognize them; you should **make it clear to the user how their data is used and retained, and who has access to it**.

Accountability

People should be accountable for AI systems. Although many AI systems seem to operate autonomously, ultimately it's the **responsibility of the developers** who trained and validated the models they use, and defined the logic that bases decisions on model predictions to **ensure that the overall system meets responsibility requirements**. To help meet this goal, designers and developers of AI-based solution should work within a framework of governance and organizational principles that ensure the solution meets responsible and legal standards that are clearly defined.

> Tip: For more information about Microsoft's principles for responsible AI, see the Microsoft responsible AI site.

Exercise - Prepare for an AI development project

Prepare for an AI development project

In this exercise, you use Azure AI Foundry portal to create a project, ready to build an AI solution.

- Open Azure AI Foundry portal
- Create a project
- Review project connections
- Test a generative AI model
- Summary: In this exercise, you've explored Azure AI Foundry, and seen how to create and manage projects and their related resources.

Module assessment

1. Which Azure resource provides language and vision services from a single endpoint? **Azure AI service**.
2. You plan to create a simple chat app that uses a generative AI model. What kind of project should you create? **Azure AI Foundry Project**.
3. Which SDK enables you to connect to resources in a project? **Azure AI Foundry SDK**.
4. How should you provide access to resources for developers who will work on multiple AI projects? **Create resource connections in an Azure AI Foundry hub**.
5. Which SDK enables you to connect to shared resources in a hub? **Azure AI Foundry SDK**.

Summary

In this module, you explored some of the key considerations when planning and preparing for AI application development. You've also had the opportunity to become familiar with **Azure AI Foundry**, the recommended platform for developing AI solutions on Azure.