

Building AI-Powered Applications with Azure Serverless Functions

Commit your Code 2025

Speaker: Naveen Chatlapalli



September 26, 2025





Agenda



What + Why + How Serverless



Azure Functions 101



AI Integration & Best Practices

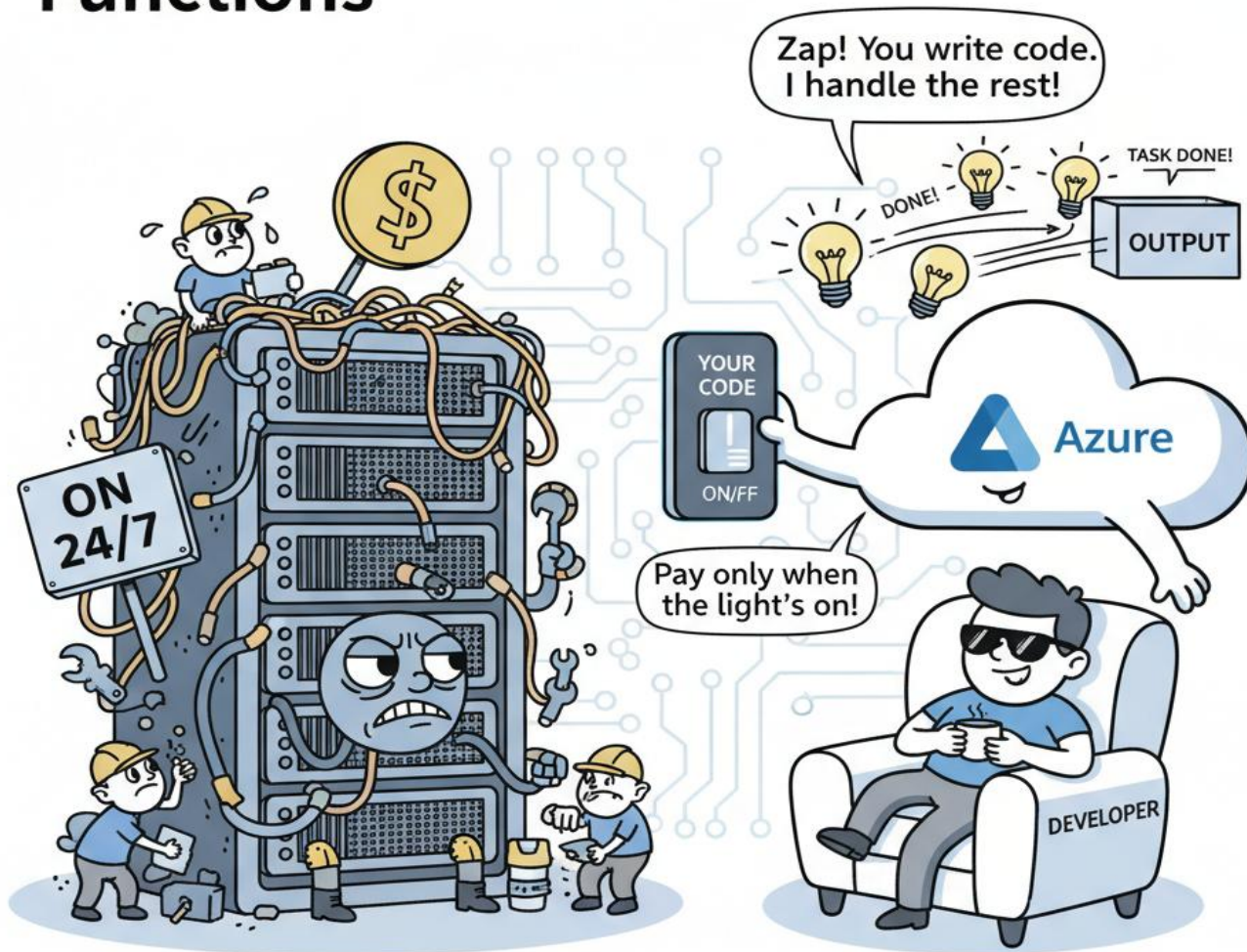


Real-world Demo



What is Serverless

Azure Serverless Functions

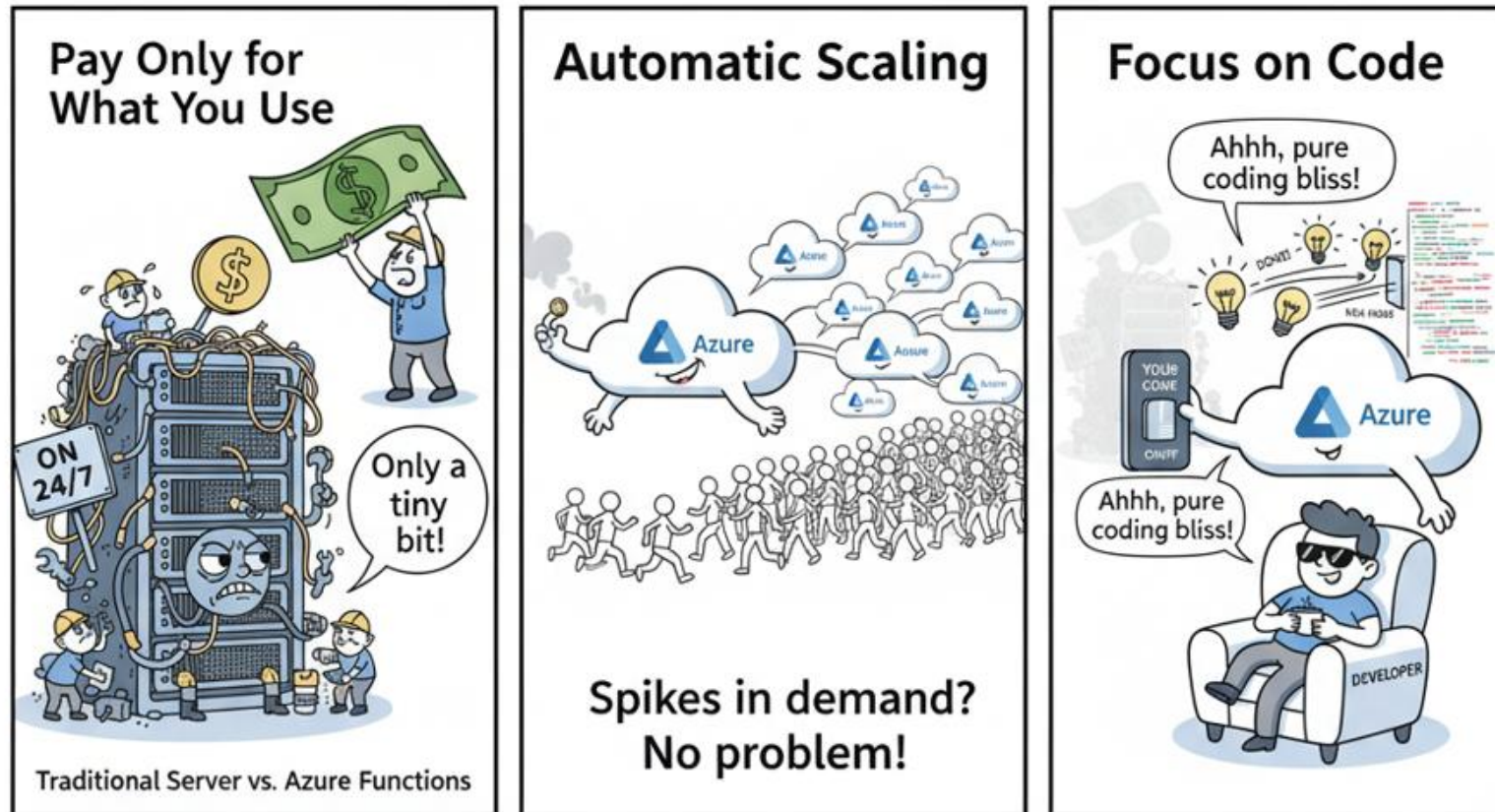


Azure Functions is a **serverless** solution that lets you run code without managing any infrastructure. You simply provide the code for what you want to do, and it only runs when triggered by an event



Why Serverless

👍 Simplicity & Cost-Efficiency



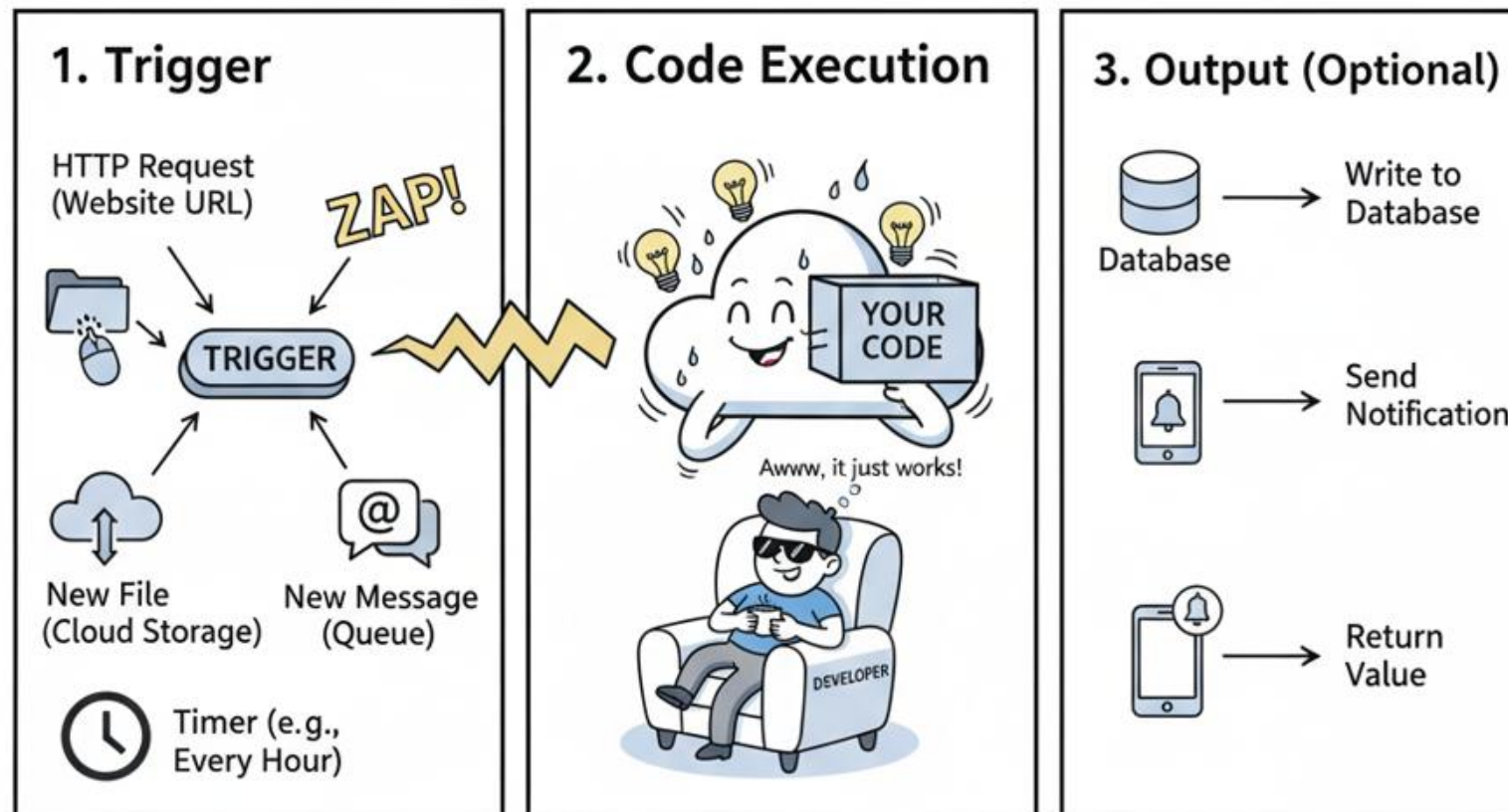
Let Azure handle the headaches!

The key benefits of using Azure Functions are **simplicity** and **cost-efficiency**. The service **automatically** scales to handle any workload, you **pay only for what you use**, and developers can **focus entirely on their code** instead of managing servers.



How Serverless

Trigger & Execute Model



Example: New Photo Uploaded → Function Resizes → Smaller Photo Saved

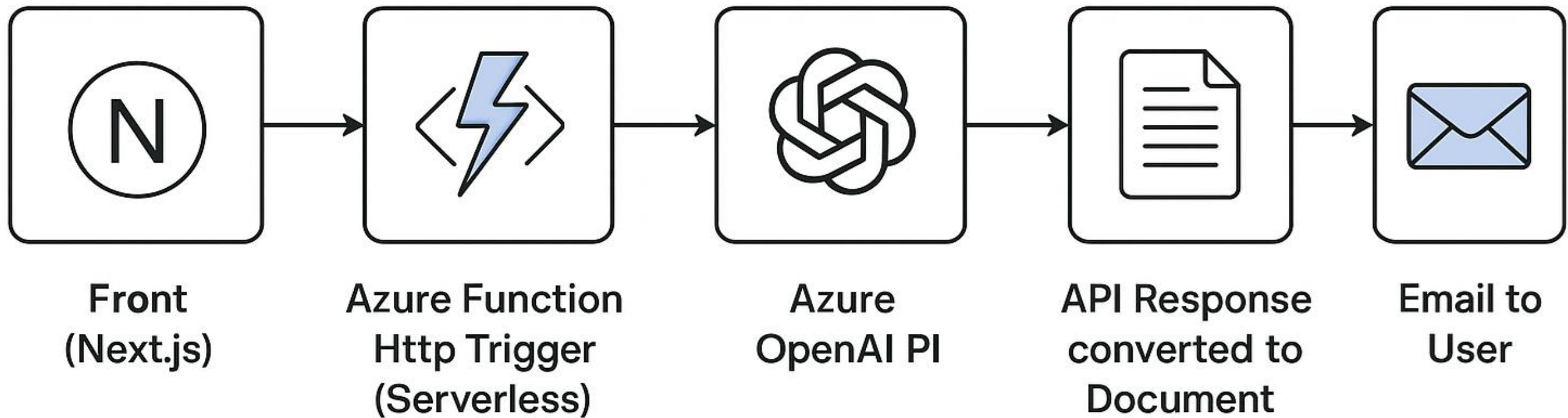
Azure Functions work using a **trigger and execute** model. An event, like a new file being uploaded or a scheduled time, acts as a **trigger** that automatically runs your code. Once executed, your function performs its task and can produce an **output**, like saving data to a database or sending a notification.





Why I chose serverless

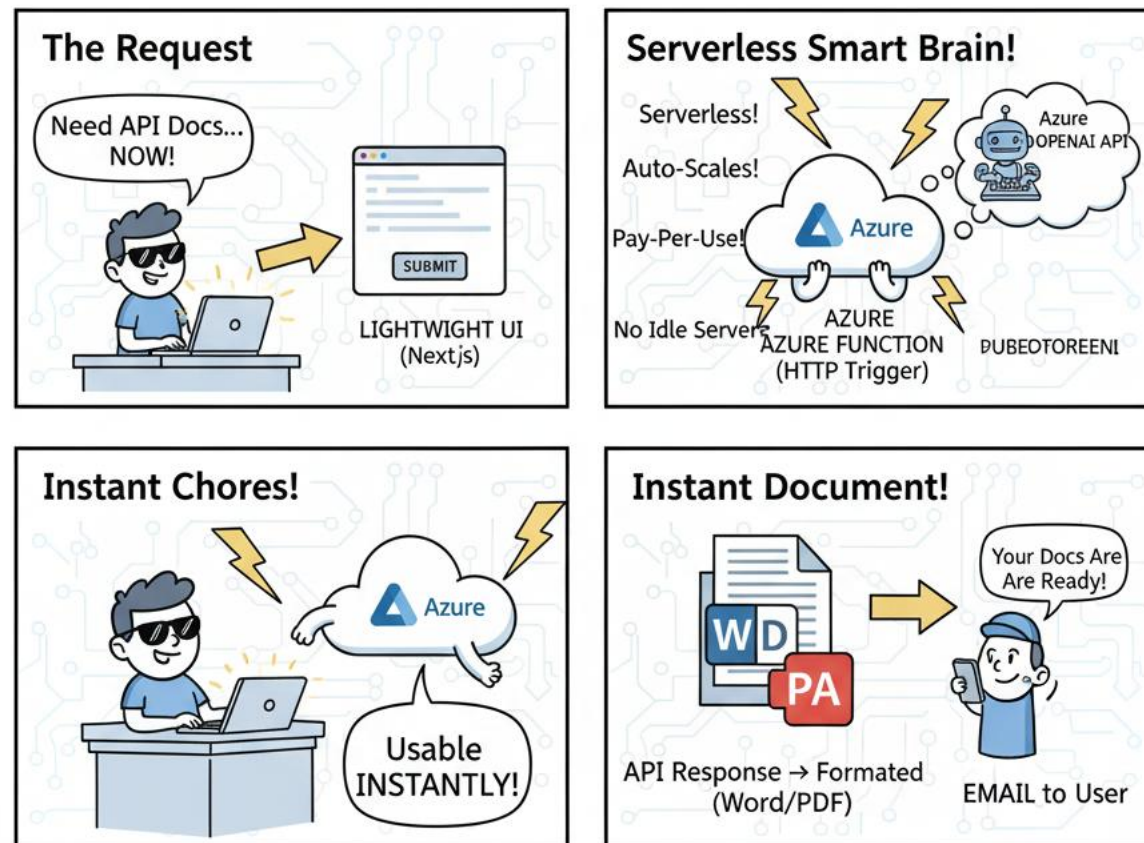
Unisloth AI-powered Documentation Agent



Product I developed

Unsloth: Your AI Documertation Agent

How it Works (The Magical Flow!)



⚡ Doc magic, powered by Unsloth & Azure! ⚡

- Serverless, scales automatically with user requests.
- Pay-per-use, so cost-efficient (no idle servers).
- Directly integrates with front-end requests via HTTPS — perfect for a request/response pattern.
- Keeps infrastructure minimal (no VM or container overhead).



Unslloth

Dashboard

Documents

Feedback

Settings

Dashboard

Beta

Chrome Extension

naveen.chatlapalli@ashlingpartners.com

Document Type

Excel

Report Type

Test Case Document

Project Description

Click the microphone to start recording

Voice Input

Example: 'Login functionality with email and password authentication, social login options, and multi-factor authentication. Includes password reset flow and session management.'

Wanna try a template?

Recipients

Add email addresses separated by semicolons

naveen.aifanatic@gmail.com

Beta



Azure Functions Overview



Triggers & Bindings

HTTP, Event Grid, Blob, Timer, and more



Hosting Plans

Consumption, Premium, Flex



Language Support

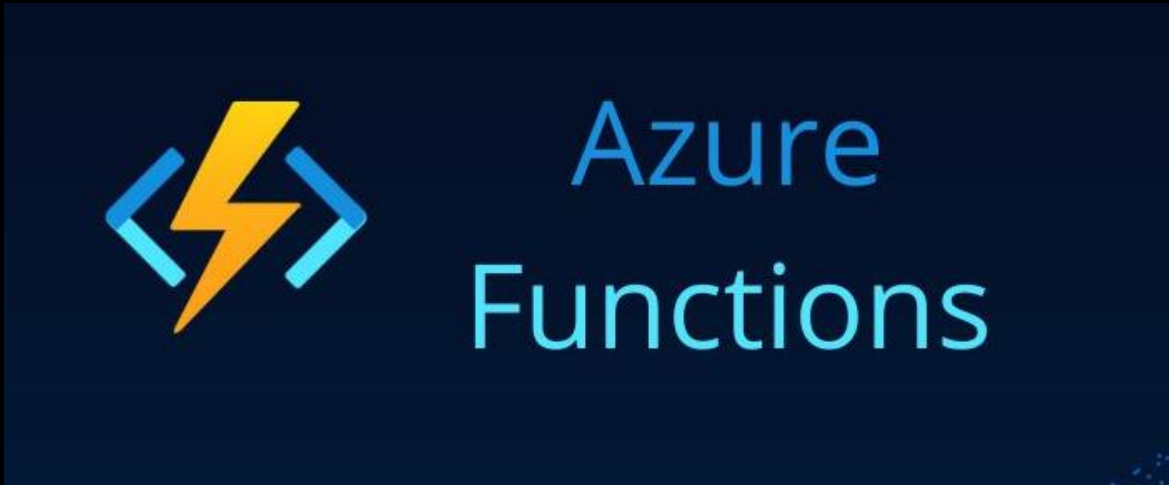
Python, C#, JavaScript, and others



DevOps Ready

Local development with func CLI, GitHub Actions deployment

Consumption vs Premium vs Flex



Plan	Cost Model	Cold Start	Key Features
Consumption	Pay per execution	Highest	Infinite scale, zero cost when idle
Premium	Reserved instances	Minimal	Always Ready, VNET support, better SLA
Flex	Consumption + burst	Medium	15-min capacity reservation, balanced approach



Event-Driven Architecture

Event Publication

System or custom events trigger workflows automatically

Fan-out Processing

Single event triggers multiple parallel functions



Event Grid Routing

CloudEvents schema standardizes message format

Event Filtering

Process only relevant events based on patterns

Blob Storage Trigger



File Upload Event

Function automatically triggers when files are uploaded to Blob Storage.



Stream Processing

Handle large files efficiently with `InputStream` binding without full download.



Batch Processing

Process multiple files in parallel with `maxBatchSize` property settings.



Orchestration

Combine with Durable Functions for complex fan-in/out processing patterns.

Best Practices for Azure Serverless



Performance

Choose right plan. Minimize cold starts. Optimize code with smart caching.



Cost

Pay only when executed. Monitor usage. Keep resources in-region.



Security

Small, focused functions. Use API Management. Apply least-privilege access.



Real-World Example: IoT sensor data triggering real-time maintenance alerts.



Azure Function – Trigger Types



Add new Azure Function and choose your trigger type

New Azure Function - EventGrid

- SQL trigger
- SQL output binding
- SQL input binding
- MySQL trigger
- Blob trigger**
- MySQL output binding
- Event Grid trigger**
- Event Grid Cloud Event trigger
- Event Hub trigger
- IoT Hub trigger
- Service Bus Topic trigger
- Durable Functions Entity
- Durable Functions Orchestration

Connection string setting name

Path

samples-workitems

☐ Configure Blob trigger connection

Updates are ready [Refresh](#)

Add Cancel

Azure Function – Event Grid Trigger Implementation



C# - Azure Function Class file which identifies the Function Name in Visual Studio

```
[FunctionName("EventBlobTrigger")]  
0 references  
public static void Run(  
    [EventGridTrigger] EventGridEvent eventGridEvent,  
    [Queue("azure-webjobs-blobtrigger-az-child-mrocfunction")] ICollector<BlobInfo> outputQueue,  
    ILogger log)...
```

After publishing the function, the Azure portal → Functions displays the same matching trigger name

Functions		Metrics	Properties	Notifications (1)
{ } Set up local environment		▼	Refresh	
Filter by name...				
Name				Trigger
BlobStorageTrigger				Blob
EventBlobTrigger				Event Grid

Azure Function – Blob Storage Trigger Implementation



C# - Azure Function Class file which identifies the Function Name in Visual Studio

```
[StorageAccount("AzureWebJobsStorage")]  
0 references  
public class BlobTrigger  
{  
    [FunctionName("BlobStorageTrigger")]  
    0 references  
    public static async Task Run([BlobTrigger("%ContainerName%")] Stream myBlob, string name, ILogger log)...
```

After publishing the function the Azure portal → Functions displays the same matching trigger name

Functions

Metrics

Properties

Notifications (1)

{ }

Set up local environment

▼

↺

Refresh

🔍

Filter by name...

Name	Trigger
BlobStorageTrigger	Blob
EventBlobTrigger	Event Grid

Azure AI Foundry - Implementation



Streamlines the entire AI development process, from exploring and choosing models to deploying and monitoring solutions at scale.

The screenshot displays the Azure AI Foundry interface for a project named 'unsloth-ai'. The left sidebar contains navigation options: Overview, Model catalog, Playgrounds, AI Services, Build and customize, Agents (PREVIEW), Templates, Fine-tuning, Content Understanding (PREVIEW), Prompt flow, Observe and optimize, Tracing (PREVIEW), Monitoring, Protect and govern, Evaluation, Guardrails + controls, Risks + alerts (PREVIEW), and Governance (PREVIEW). The main content area is divided into three sections:

- Endpoints and keys:** Contains an 'API Key' field, which is highlighted by an orange arrow. A 'View all endpoints' link is also present.
- Included capabilities:** Lists 'Azure AI inference', 'Azure OpenAI', and 'Azure AI Services'. Under 'Azure AI inference', the 'Azure AI model inference endpoint' (marked as PREVIEW) is highlighted by an orange arrow. The endpoint URL is `https://ai-unslothhub973643316846.services.ai.azure.com/m`. A link for '{x} API documentation' is also visible.
- Project details:** Displays the 'Project connection string', 'Subscription' (Ashling Partners - Dev), 'Subscription ID' (3883633c-4e4e-4bef-8355-988a1426c3e9), and 'Location' (northcentralus). It also includes a 'Manage project settings' section with options to 'Add users', 'View quota', 'Connect resources', and 'Track costs', along with an 'Open in management center' button.

Azure AI Foundry – Deploy AI Model



GPT-4o mini → is the most cost-efficient small model and has vision capabilities. The model has 128K context and an October 2023 knowledge cutoff.

Cost Per 1M Token: Input: \$0.15 → Output: \$0.60

Manage deployments of your models and services

[Model deployments](#) [Service endpoints](#)

+ Deploy model ▾

↻ Refresh

✎ Edit

🔗 Open in playground

↺ Reset view

Name	Model name	Model version	State	Model retirement date	Content filter
^ ai-unslottHub973643316846_a0ai <small>Azure AI Services</small> <div>🔗 Get endpoint</div>					
<div>○</div> <div>gpt-4o-mini</div>	gpt-4o-mini	2024-07-18	Succeeded		DefaultV2 ⓘ

Azure AI Foundry – Deploy AI Model



GPT-4o mini → is the most cost-efficient small model and has vision capabilities. The model has 128K context and an October 2023 knowledge cutoff.

Cost Per 1M Token: Input: \$0.15 → Output: \$0.60

Manage deployments of your models and services

[Model deployments](#) [Service endpoints](#)

+ Deploy model ▾

↻ Refresh

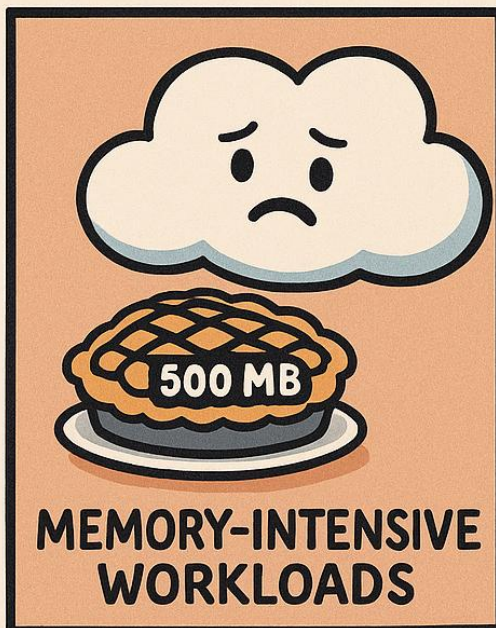
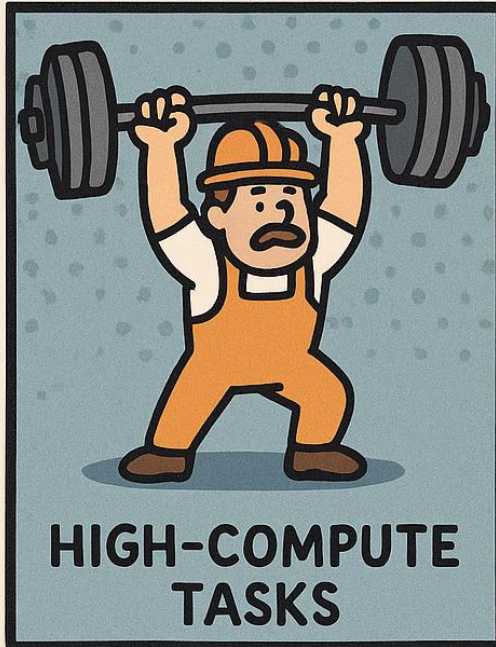
✎ Edit

🔗 Open in playground

↺ Reset view

	Name	Model name	Model version	State	Model retirement date	Content filter
^	ai-unslottHub973643316846_a0ai	Azure AI Services	Get endpoint			
○	gpt-4o-mini	gpt-4o-mini	2024-07-18	Succeeded		DefaultV2 ⓘ

WHEN NOT TO USE AZURE SERVERLESS FUNCTIONS

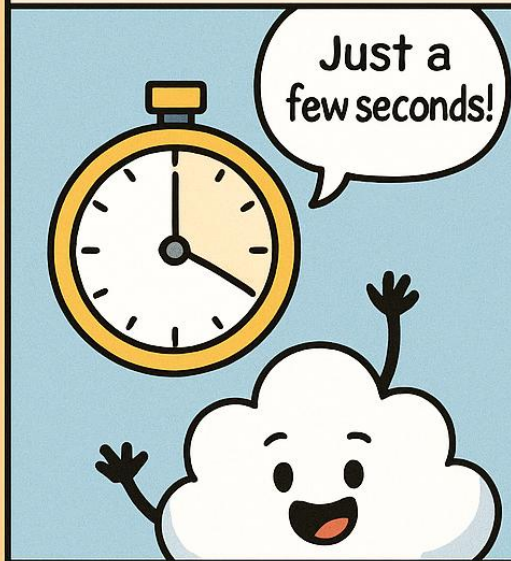


Azure Functions are not ideal for every situation. You should avoid using them for **long-running processes** or constant **high-compute tasks** that require sustained power. They are also a poor choice for workloads that are very **memory-intensive** or involve slow operations, as serverless platforms have built-in time and resource limits.

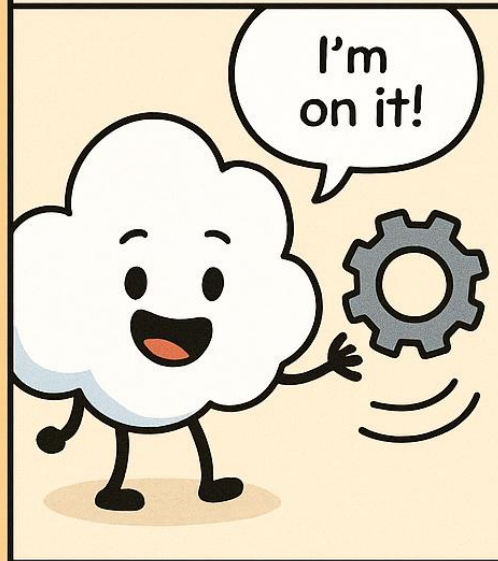


BEST USE CASES FOR AZURE SERVERLESS FUNCTIONS

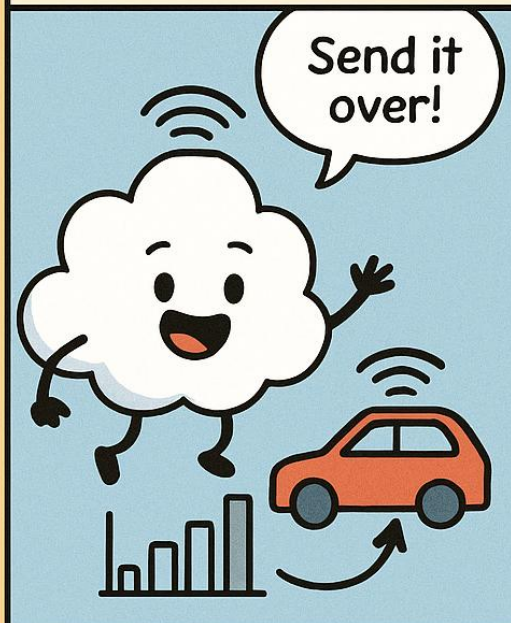
SHORT-LIVED TASKS



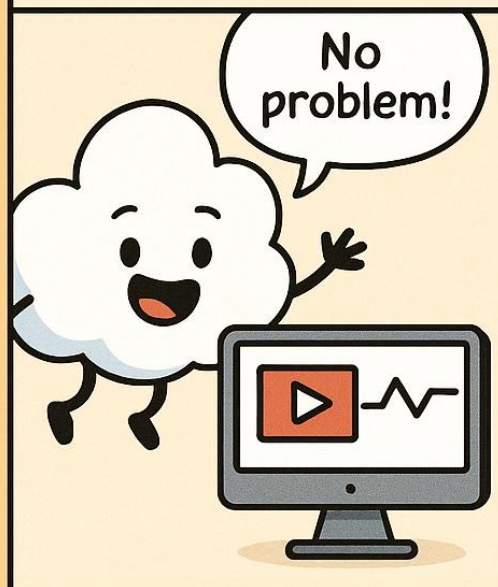
BACKGROUND JOBS



IoT DATA PROCESSING



REAL-TIME STREAM HANDLING



Azure Functions are ideal for **short-lived, automated tasks** that run in response to an event. They are perfect for running **background jobs**, such as sending a confirmation email, or for processing continuous streams of data from **IoT devices** and **real-time applications**.



Contact Me!

Website: <https://naveen.aifanatic.pro>

Email: naveen.aifanatic@gmail.com