# Using .NET(Csharp) Implemented Data Transfer - Azure Blob Storage to Web Application.

## **Problem Statement: -**

- Our Company firstly use one specific server or database server to store files.
- We can upload only specific size of files. If we upload more then server file size then Getting error-file storage has been full.
- When we use database server: -
  - Massive increase in traffic between the web server and the database
  - Slowdown in database performance
  - Massive increase in the size of the database, resulting in a backup headache
  - If you store docs outside of the database you **will** have missing documents and broken links soomer not later. Your backup/restore scenario is a lot more complex: you have no way to ensure that all data is from the same point in time.

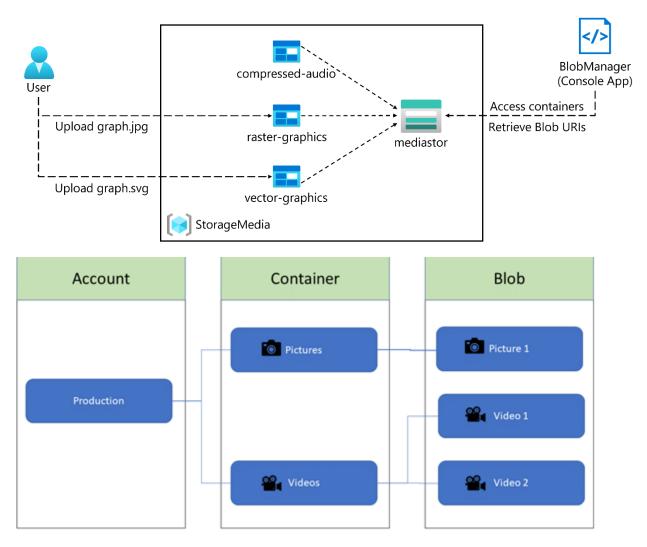
#### **Introduction: -**

Microsoft offers several options to store data on the cloud. Each option has its unique purpose for serving different business needs. One of the significant capabilities that Microsoft Azure provides is the agility to migrate to different storage options if required.

There are various options available in the azure storage account for storing user data.

- Blob Storage
- File Storage
- Table Storage
- Queue Storage
- Disk

# Architecture Diagram :



### What is Azure Blob Storage?

Azure Blob Storage is an object storage solution for the cloud. Blob Storage allows you to store a massive amount of unstructured data. The unstructured data need not be of the specific data model.

### What is Azure Blob Storage used for?

Azure Blob Storage was designed to serve specific needs. If your business use case needs to store unstructured data like audio, video, images, etc then you should probably go with this option. The objects which are being stored in Blob does not necessarily have an extension. The following points describe the use case scenarios:

- Serving images or documents directly to a browser
- Storing Files for distributed access
- Streaming video and audio
- Writing to log Files
- Storing data for backup, restore, disaster recovery and archiving
- Storing data for analysis by an on-premises or Azure-hosted service

## How does BLOB storage work?

Blob Storage comprises of three different types of resources:

- 1. The Storage Account
- 2. A **container** in the storage account
- 3. A **blob** in a container

binary data such as video, audio, and images.

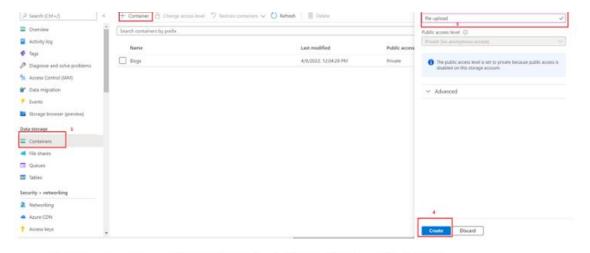
Create a Azure Storage resource using Azure Portal :-

■ Microsoft Azure	D Search resources, services, and docs (G+/)
Home > Create a resource >	
Storage account 👒	
Microsoft	
Storage acc	count 👳 Add to Favorites
Microsoft	
★ 4.2 (1749 Azure rat	ings)
Plan	
Storage account	✓ Create
Overview Plans Usage Inf	formation + Support Reviews
	durable cloud storage, backup, and recovery solutions for any data, big or small. It works with the infrastructure you already have to cost-

anced	Networking	Data protection	Encryption	Tags	Review + create	
ils						
			unt. Choose a ne	ew or exist	ing resource group to organi	ze and
		Visual Studio Prof	essional			~
ce group		azuretraining				~
		Create new				
nt name	•					
		(US) East US				~
0 *		Standard: Rec	ommended for r	most scena	rios (general-purpose v2 acc	ount)
		O Premium: Rec	ommended for s	scenarios t	hat require low latency.	
i *		Geo-redundant st	orage (GRS)			~
0		<b>1 1 1 1</b>		able is the	event of regional unavailabil	<b>1</b>
	scription storage a rce group <b>ails</b> create a nt name	scription in which to create storage account together v rce group * ails create a legacy storage ac nt name ① *	scription in which to create the new storage account together with other resources.          Visual Studio Prof         rce group *       azuretraining         ails         create a legacy storage account type, please click         nt name ① *         (US) East US         ① *         ① *	scription in which to create the new storage account. Choose a n storage account together with other resources. Visual Studio Professional tree group * azuretraining Create new ails create a legacy storage account type, please click here. nt name ① * (US) East US ① *	scription in which to create the new storage account. Choose a new or exist storage account together with other resources. Visual Studio Professional rce group * azuretraining Create new ails create a legacy storage account type, please click here. nt name ① * (US) East US ① *	scription in which to create the new storage account. Choose a new or existing resource group to organi storage account together with other resources.          Visual Studio Professional         cce group *       azuretraining         create new         ails         create a legacy storage account type, please click here.         nt name ① *         (US) East US

≡ Microsoft Azure	, $\rho$ Search resources, services, and docs (G+/)			Σ	Ģ	ø	0	D &
Home > filestgaccount_ Deployment	1649486015501   Overview 🖉							
P Search (Ctrl+/)	🔹 🖹 Delete 🚫 Cancel 🏥 Redeploy	C Refresh						
🙏 Overview	$\bigcirc \ \ {\rm We'd \ love \ your \ feedbackl} \rightarrow$							
Dutputs	••• Deployment is in pro-	gress						
E Template	Deployment name: filestgaccount_t Subscription: Visual Studio Professio Resource group: azuretraining	649486015501 mal	Start time: 4/9/2022.12:03:50 PM Correlation ID: f8a83339-1a25-4fb8-af40-612007e9ef81					
	∧ Deployment details (Download)							
	Resource	Type	Status				Ope	ration details
	No results.							

filestgaccount	-					×
,P Search (Ctrl+/)	🛪 👌 Open in Explorer 🔋 Delete	$r \rightarrow Move \sim \bigcirc$ Refresh 📱 Mobile 👰 Fr	redback			
Cverview	* ^ Essentials					250N View
Activity log	Resource group (minut) : ature	etraining	Performance : Sta	ndard		
Tags	Location : East	US	Replication : Rea	id-access geo-redundant	t storage (RA-GRS)	
Diagnose and solve problems	Primary/Secondary Location : Prima	ary: East US, Secondary: West US	Account kind : Sto	rageV2 (general purpose	r vZ)	
Access Control (IAM)	Subscription (move) ! Visual	al Studio Professional	Provisioning state 1 Suc	zeeded		
Data migration	Subscription ID :		Created = 4/9	(2022, 12:03:59 PM		
A REAL PROPERTY AND A REAL	Disk state : Prim	ary: Available, Secondary: Available				
Events	Tags (edit) : Click here to	add tags				
Storage browser (preview)						
Data storage	Properties Monitoring Co	apabilities (7) Recommendations Tutorials	Developer Tools			
Containers	Elob service		Security			
File shares	Hierarchical namespace	Disabled	Require secure tran	sfer for REST API	Enabled	
Queues	Default access tier	Hot	operations			
	Blob public access	Disabled	Storage account ke	ry access	Enabled	
Tables						
	Blob soft delete	Enabled (7 days)	Minimum TLS versi		Version 1.2	
Tables Security + networking		Enabled (7 days) Enabled (7 days)	Minimum TLS versi Infrastructure encry		Version 1.2 Disabled	
	Blob soft delete		Infrastructure encr			
iecurity + networking	Blob soft delete Container soft delete	Enabled (7 days)		ption		



Next step is to create an Azure function that uploads files into the "file-upload" container.

earch (Ch1+/) ×	+ Container 👌 Change access level 🤊 Restore	containen 🗸 🕐 Refresh 🕴 🗐 Deiete		
Overview	Search containers by prefix			Show deleted containers
Activity log	Name	Last modified	Public access level	Lease state
lage	Slogs	4/9/2022, 12:04:29 PM	Private	Available
lagnose and solve problems	Re-upload	4/9/2022, 12:14:11 PM	Private	Available
ccess Control (IAM)				
era migration				
torage browser (preview)				

```
Code :
```

```
{
    "IsEncrypted": false,
    "Values": {
        "AzureWebJobsStorage": "<replace your blob storage connection
key here>",
        "ContainerName": "file-upload", // Container name
        "FUNCTIONS WORKER_RUNTIME": "dotnet"
}
}
using Azure.Storage.Blobs;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Azure.WebJobs;
using Microsoft.Azure.WebJobs.Extensions.Http;
using Microsoft.Extensions.Logging;
using System;
using System.IO;
using System.Threading.Tasks;
namespace FileUploadFunction {
    public static class FileUpload {
        [FunctionName("FileUpload")]
        public static async Task < IActionResult > Run(
            [HttpTrigger(AuthorizationLevel.Anonymous, "post", Route =
null)] HttpRequest req, ILogger log) {
            string Connection =
Environment.GetEnvironmentVariable("AzureWebJobsStorage");
            string containerName =
Environment.GetEnvironmentVariable("ContainerName");
            Stream myBlob = new MemoryStream();
            var file = reg.Form.Files["File"];
            myBlob = file.OpenReadStream();
            var blobClient = new BlobContainerClient(Connection,
containerName);
            var blob = blobClient.GetBlobClient(file.FileName);
            await blob.UploadAsync(myBlob);
            return new OkObjectResult("file uploaded successfylly");
```

```
}
}
using Azure.Storage.Blobs;
using Azure.Storage.Blobs.Models;
using Azure.Identity;
// TODO: Replace <storage-account-name> with your actual storage account name
var blobServiceClient = new BlobServiceClient(
        new Uri("https://<storage-account-name>.blob.core.windows.net"),
       new DefaultAzureCredential());
//Create a unique name for the container
string containerName = "quickstartblobs" + Guid.NewGuid().ToString();
// Create the container and return a container client object
BlobContainerClient containerClient = await
blobServiceClient.CreateBlobContainerAsync(containerName);
// Create a local file in the ./data/ directory for uploading and downloading
string localPath = "data";
Directory.CreateDirectory(localPath);
string fileName = "quickstart" + Guid.NewGuid().ToString() + ".txt";
string localFilePath = Path.Combine(localPath, fileName);
// Write text to the file
await File.WriteAllTextAsync(localFilePath, "Hello, World!");
// Get a reference to a blob
BlobClient blobClient = containerClient.GetBlobClient(fileName);
Console.WriteLine("Uploading to Blob storage as blob:\n\t {0}\n",
blobClient.Uri);
// Upload data from the local file
await blobClient.UploadAsync(localFilePath, true);
Console.WriteLine("Listing blobs...");
// List all blobs in the container
await foreach (BlobItem blobItem in containerClient.GetBlobsAsync())
{
   Console.WriteLine("\t" + blobItem.Name);
}
```

```
// Download the blob to a local file
// Append the string "DOWNLOADED" before the .txt extension
// so you can compare the files in the data directory
string downloadFilePath = localFilePath.Replace(".txt", "DOWNLOADED.txt");
Console.WriteLine("\nDownloading blob to\n\t{0}\n", downloadFilePath);
// Download the blob's contents and save it to a file
await blobClient.DownloadToAsync(downloadFilePath);
// Clean up
Console.Write("Press any key to begin clean up");
Console.ReadLine();
Console.WriteLine("Deleting blob container...");
await containerClient.DeleteAsync();
Console.WriteLine("Deleting the local source and downloaded files...");
File.Delete(localFilePath);
File.Delete(downloadFilePath);
```

```
Console.WriteLine("Done");
```

**Challenges Faced :** 

Integration complexity: Syncing Azure Blob Storage with a .NET application requires careful integration.

Data consistency: Ensuring data consistency between the application and Blob Storage can be challenging.

Scalability considerations: Handling growing data volumes while maintaining performance is a challenge.

Error handling and logging: Robust error handling and logging mechanisms are necessary for troubleshooting.

Security and access control: Managing access control and security during synchronization can be complex.

**Business Benefit** :-

- Perfect for small businesses and established enterprises Azure is designed for every business regardless of size, from the local bakery to multi-national corporations. It's easily scalable to meet your IT demands and operates on a pay-as-you-go pricing model to meet any budget. Since businesses can launch and store internal and external applications in the cloud, it also saves on in-house IT costs, including hardware and maintenance.
- Complements and expands your current IT infrastructure Azure allows your IT personnel to focus on your business without having to worry about in-house capabilities or maintaining equipment that is over-taxed or underused. The platform makes it fast and easy to deploy your current apps with little to no downtime. An integrated development environment reduces the learning curve, allowing teams to master the platform quickly. Additionally, the platform has a footprint in more countries than Google or Amazon, providing faster content delivery while optimizing the user experience. Azure is scalable to grow with your company, and you pay for only what you need.
- Leading the way with IaaS and PaaS At the forefront of IaaS and PaaS, Azure offers rapid deployments. The hybrid cloud environment allows companies to select whether they operate autonomously or utilize a public cloud. You are also able to decide the level at which you are connected to the internet, if at all. Meet all your IT and service needs without having to maintain the underlying infrastructure.
- Security, compliance, and disaster recovery Microsoft understands the importance of security and has designed Azure to stay ahead of the competition when it comes to protecting your data. Azure has many compliance certifications and is a top choice of high-risk industries such as health care and government to provide cloud services. Both the platform and end users are protected. Additional services such as multi-factor authentication and sophisticated disaster recovery abilities that can restore data in a matter of hours further address business needs.

Industry-specific applications — Due to the high-risk and sensitive nature of certain industries, Azure has designed specific applications to address unique needs.
 Government, health care, manufacturing, and financial services benefit greatly from Azure's many features, including offline cloud services, individualized security needs, simplified compliance, and modernized customer apps.

#### **References**:

- Azure RBAC action: <u>Microsoft.Storage/storageAccounts/blobServices/containers/blobs/write</u> (for writing to an existing blob) or <u>Microsoft.Storage/storageAccounts/blobServices/containers/blobs/add/action</u> (for writing a new blob to the destination)
- <u>Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read</u>

- by Priyanka Dingankar Senior Full Stack Developer