

# ShareSafely - File Share Web App

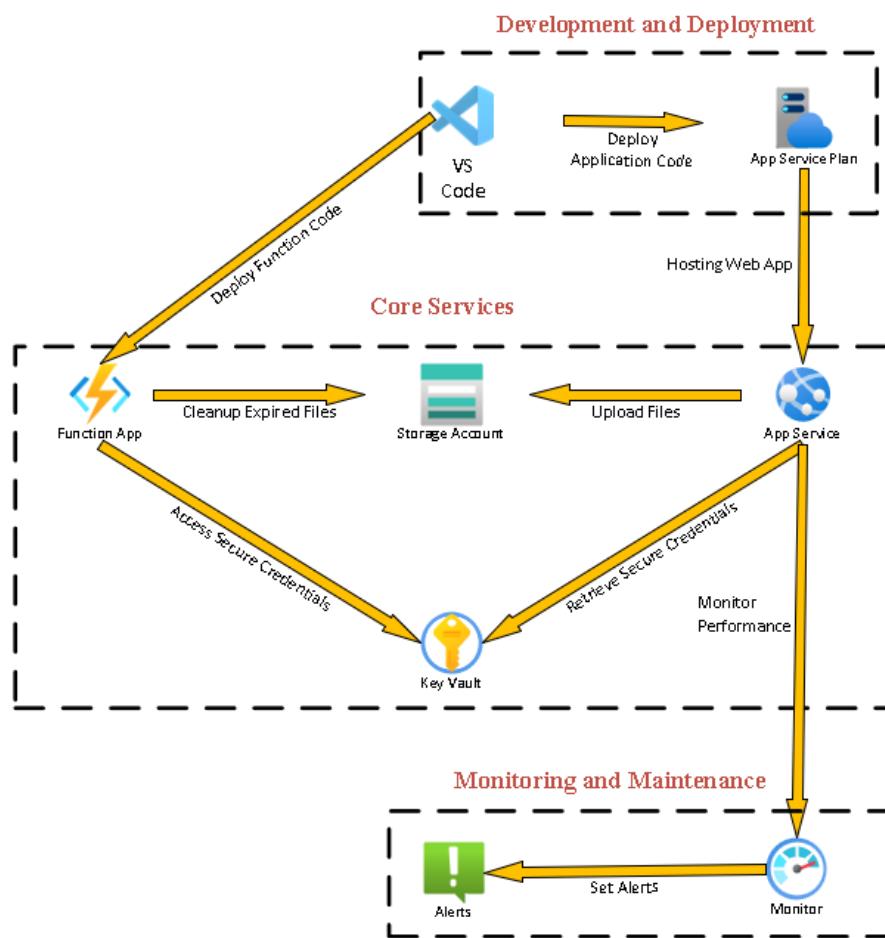
## Detailed Documentation with Screenshots

---

### Table of Contents

1. Introduction
2. Azure Blob Storage Setup
3. Web Application Development
4. Secure Credentials with Azure Key Vault
5. Deployment to Azure Web Apps
6. Monitoring and Maintenance
7. Azure Functions for Cleanup
8. Testing the Project
9. Challenges and Learnings
10. Conclusion

## Project Workflow



---

## Introduction

Welcome to the documentation for ShareSafely, a web application developed to securely manage file uploads using Azure Blob Storage. ShareSafely allows users to upload files and generate unique, time-limited links for secure sharing, ensuring that only authorized individuals can access the shared files for a specified duration. This system enhances security and control over file access.

In addition to secure file sharing, ShareSafely includes an automated mechanism for cleaning up expired files using Azure Functions, helping maintain efficient and secure storage management.

NOTE: All Scripts are in my Github Repo:

<https://github.com/vivekvashisht04/ShareSafely>

---

## Azure Blob Storage Setup

**Description:** Setting up Azure Blob Storage for storing uploaded files.

**Steps:**

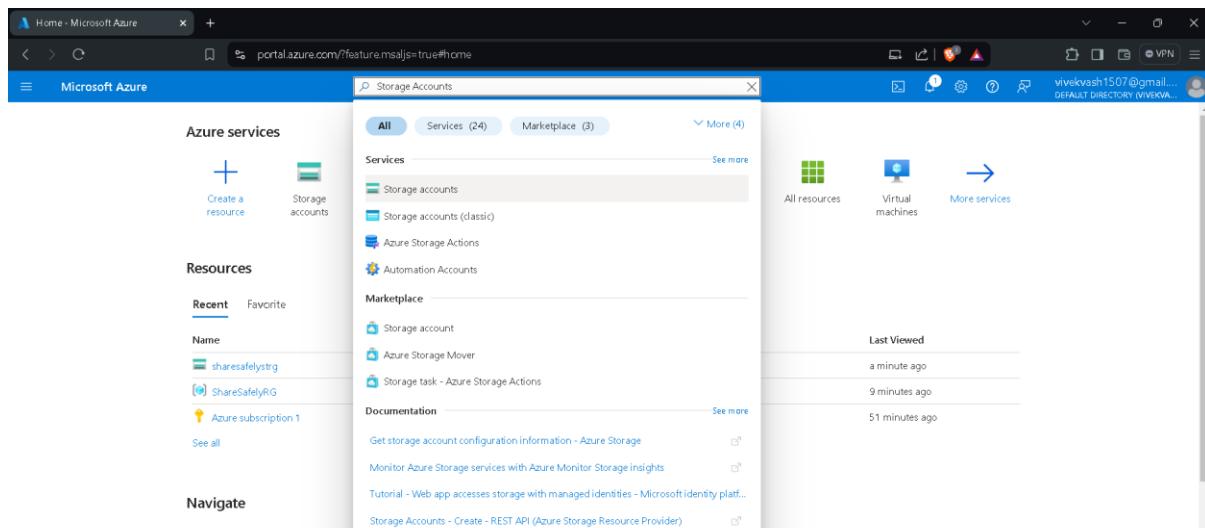
1. **Created Storage Account:**

- Navigated to the Azure Portal and create a storage account named sharesafelystrg.

2. **Configured Container:**

- Created a container named uploads and configure security settings:
  - Secure transfer required: Enabled
  - Allow Blob anonymous access: Disabled
  - Soft delete for blobs and containers: Enabled (7 days retention)

**Screenshots:**



**Create a storage account**

**Review + create**

**Basics**

Subscription	Azure subscription 1
Resource group	ShareSafetyRG
Location	East US
Storage account name	sharesafelystrg
Performance	Standard
Replication	Locally-redundant storage (LRS)

**Advanced**

Enable hierarchical namespace	Disabled
Enable SFTP	Disabled
Enable network file system v3	Disabled
Allow cross-tenant replication	Disabled
Access tier	Hot
Enable large file shares	Enabled

**Security**

Previous Next Create Give feedback

**Create a storage account**

**Security**

Secure transfer	Enabled
Blob anonymous access	Disabled
Allow storage account key access	Enabled
Default to Microsoft Entra authorization in the Azure portal	Disabled
Minimum TLS version	Version 1.2
Permitted scope for copy operations (preview)	From any storage account

**Networking**

Network connectivity	Public endpoint (all networks)
Default routing tier	Microsoft network routing

**Data protection**

Point-in-time restore	Disabled
Blob soft delete	Enabled
Blob retention period in days	7
Container soft delete	Enabled
Container retention period in days	7
File share soft delete	Enabled

Previous Next Create Give feedback

A Create a storage account - Microsoft Azure

portal.azure.com/?feature.msals=true#create/Microsoft.StorageAccount-ARM

Microsoft Azure

Search resources, services, and docs (G+)

vivekvash1507@gmail.com

DEFAULT DIRECTORY (VIVEKVA...)

Home > Storage accounts >

## Create a storage account

**Networking**

Network connectivity	Public endpoint (all networks)
Default routing tier	Microsoft network routing

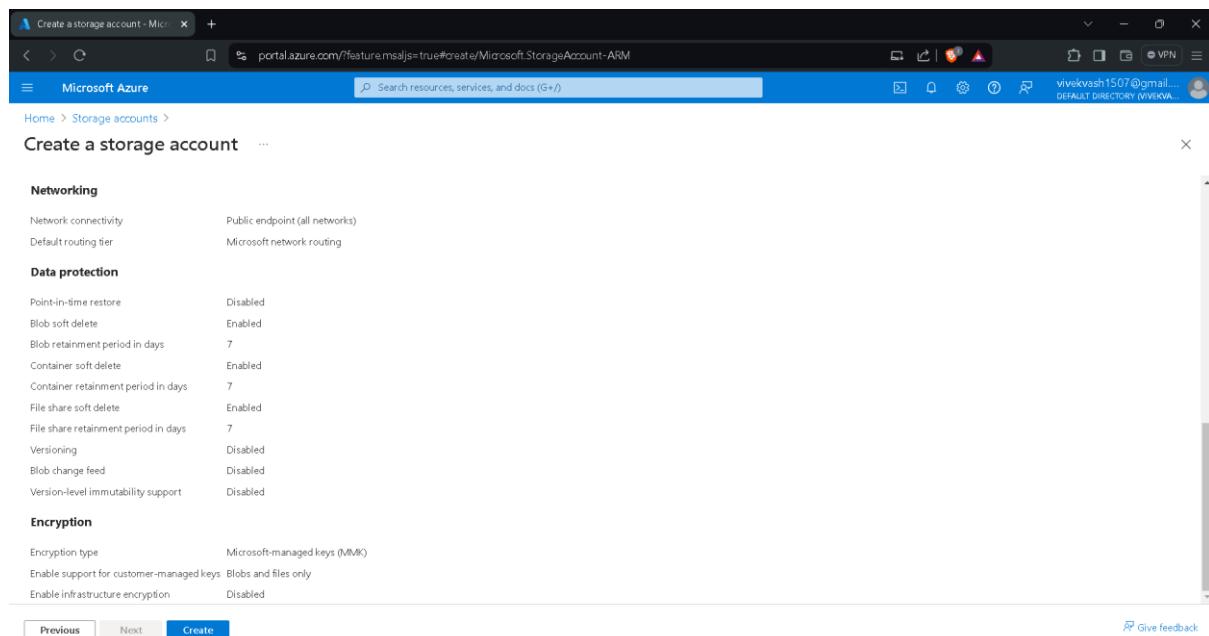
**Data protection**

Point-in-time restore	Disabled
Blob soft delete	Enabled
Blob retention period in days	7
Container soft delete	Enabled
Container retention period in days	7
File share soft delete	Enabled
File share retention period in days	7
Versioning	Disabled
Blob change feed	Disabled
Version-level immutability support	Disabled

**Encryption**

Encryption type	Microsoft-managed keys (MVK)
Enable support for customer-managed keys	Blobs and files only
Enable infrastructure encryption	Disabled

Previous Next Create Give feedback



A sharesafelystrg\_1721934091817 - Microsoft Azure

portal.azure.com/?feature.msals=true#view/HubsExtension/DeploymentDetailsBlade/~ /overview/d/92fsubscriptions%2Fee9ea1...</p><p>Microsoft Azure</p><p>Search resources, services, and docs (G+)</p><p>vivekvash1507@gmail.com<br/>DEFAULT DIRECTORY (VIVEKVA...)</p><p>Home > Deployment > sharesafelystrg\_1721934091817 | Overview </p><h2>Your deployment is complete</h2><table border="1"><thead><tr><th>Resource</th><th>Type</th><th>Status</th><th>Operation details</th></tr></thead><tbody><tr><td>sharesafelystrg/default</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr><tr><td>sharesafelystrg/default</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr><tr><td>sharesafelystrg</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr></tbody></table><h3>Deployment details</h3><table border="1"><thead><tr><th>Resource</th><th>Type</th><th>Status</th><th>Operation details</th></tr></thead><tbody><tr><td>sharesafelystrg/default</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr><tr><td>sharesafelystrg/default</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr><tr><td>sharesafelystrg</td><td>Microsoft.Storage/storageAccounts</td><td>OK</td><td>Operation details</td></tr></tbody></table><h3>Next steps</h3><button>Go to resource</button><div>Give feedback<br/>Tell us about your experience with deployment</div><div>Cost Management<br/>Get notified to stay within your budget and prevent unexpected charges on your bill.<br/>Set up cost alerts ></div><div>Microsoft Defender for Cloud<br/>Secure your apps and infrastructure<br/>Go to Microsoft Defender for Cloud ></div><div>Free Microsoft tutorials<br/>Start learning today ></div><div>Work with an expert<br/>Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.<br/>Find an Azure expert ></div>

sharesafelystrg - Microsoft Azure

Microsoft Azure

sharesafelystrg | Configuration

Storage account

Search Save Discard Refresh Give feedback

The cost of your storage account depends on the usage and the options you choose below. [Learn more about storage pricing](#)

Account kind: StorageV2 (general purpose v2)

Performance: Standard (radio button selected) Premium

This setting cannot be changed after the storage account is created.

Secure transfer required: Enabled (radio button selected) Disabled

Allow Blob anonymous access: Enabled (radio button selected) Disabled

Allow storage account key access: Enabled (radio button selected) Disabled

Allow recommended upper limit for shared access signature (SAS) expiry interval: Enabled (radio button selected) Disabled

Default to Microsoft Entra authorization in the Azure portal: Enabled (radio button selected) Disabled

Minimum TLS version: Version 1.2

https://go.microsoft.com/fwlink/?LinkId=331014

Permitted scope for CORS operations (read/write):

This screenshot shows the 'Configuration' tab for a Storage account named 'sharesafelystrg'. It includes sections for account kind (StorageV2), performance (Standard selected), secure transfer (Enabled), blob anonymous access (Enabled), storage account key access (Enabled), SAS expiry interval (Enabled), Microsoft Entra authorization (Enabled), and TLS version (Version 1.2). A sidebar lists other storage-related configurations like Data Lake Gen2 upgrade, Resource sharing (CORS), Advisor recommendations, and endpoints.

sharesafelystrg - Microsoft Azure

Microsoft Azure

sharesafelystrg | Data protection

Storage account

Search

Shared access signature

Encryption

Microsoft Defender for Cloud

Data management

Storage tasks (preview)

Redundancy

Data protection (selected)

Object replication

Blob inventory

Static website

Lifecycle management

Azure AI Search

Settings

Configuration

Data Lake Gen2 upgrade

Resource sharing (CORS)

Advisor recommendations

Recovery

Enable Azure Backup for blobs

Enable point-in-time restore for containers

Enable soft delete for blobs (checkbox selected)

Soft delete enables you to recover blobs that were previously marked for deletion, including blobs that were overwritten. [Learn more](#)

Keep deleted blobs for (in days): 7

Enable soft delete for containers (checkbox selected)

Soft delete enables you to recover containers that were previously marked for deletion. [Learn more](#)

Keep deleted containers for (in days): 7

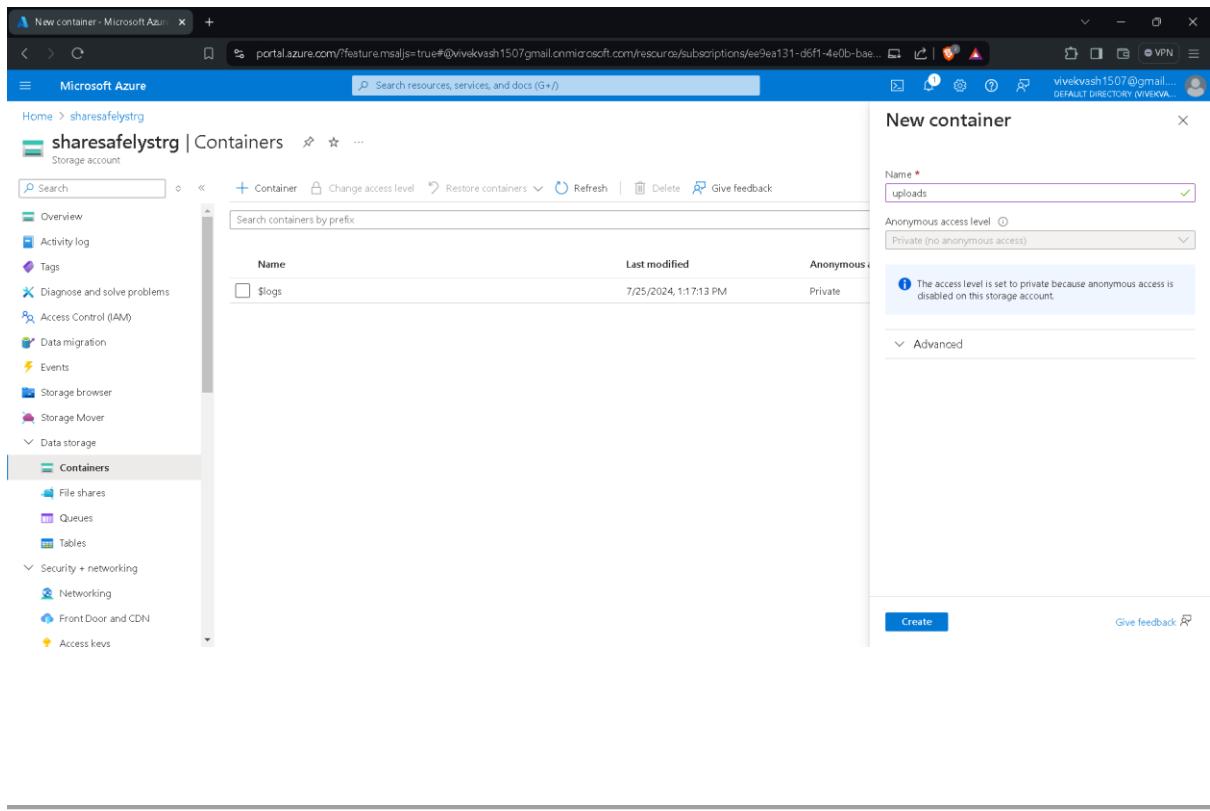
Enable permanent delete for soft deleted items

Tracking

Enable versioning for blobs

Save Discard Give feedback

This screenshot shows the 'Data protection' tab for the same Storage account. It includes sections for recovery (Azure Backup for blobs, point-in-time restore, and soft delete for blobs and containers) and tracking (blob versioning). The sidebar also lists other data protection features like shared access signatures, encryption, and Microsoft Defender for Cloud.



## Web Application Development

### Overview

In this section, I outlined the process of developing the "ShareSafely" web application, focusing on setting up the development environment, coding the application, and testing it locally.

Note: All Scripts are provided in my GitHub Repo

### Development Environment Setup

**Objective:** To set up the necessary tools and environment for developing the web application in Python.

#### 1. Programming Language: Python

- **Background:** I chose Python due to my prior experience from high school Computer Science courses and ITS Diploma courses, which included Object-Oriented Programming (OOP) and Scripting.

#### 2. Source-Code Editor: Visual Studio Code (VS Code)

- **Extensions Installed:**
  - Python Extension

### 3. Project Initialization:

- **Project Folder:** Created a directory named ShareSafelyApp.
- **Virtual Environment:**
  - Command: python -m venv venv
- **Activation:**
  - Command: .\venv\Scripts\activate
- **Package Installation:**
  - Command: pip install Flask azure-storage-blob azure-identity azure-functions

## Project Structure

The project follows a structured format to separate code and resources effectively:

```
arduino
Copy code
ShareSafelyApp/
    ├── app.py
    ├── templates/
        ├── upload.html
        └── link.html
    └── static/
        └── styles.css
```

## Files and Directories:

- **app.py:** Main application logic
- **templates/:** HTML templates for file upload and link generation
- **static/:** CSS for styling

## Key Components:

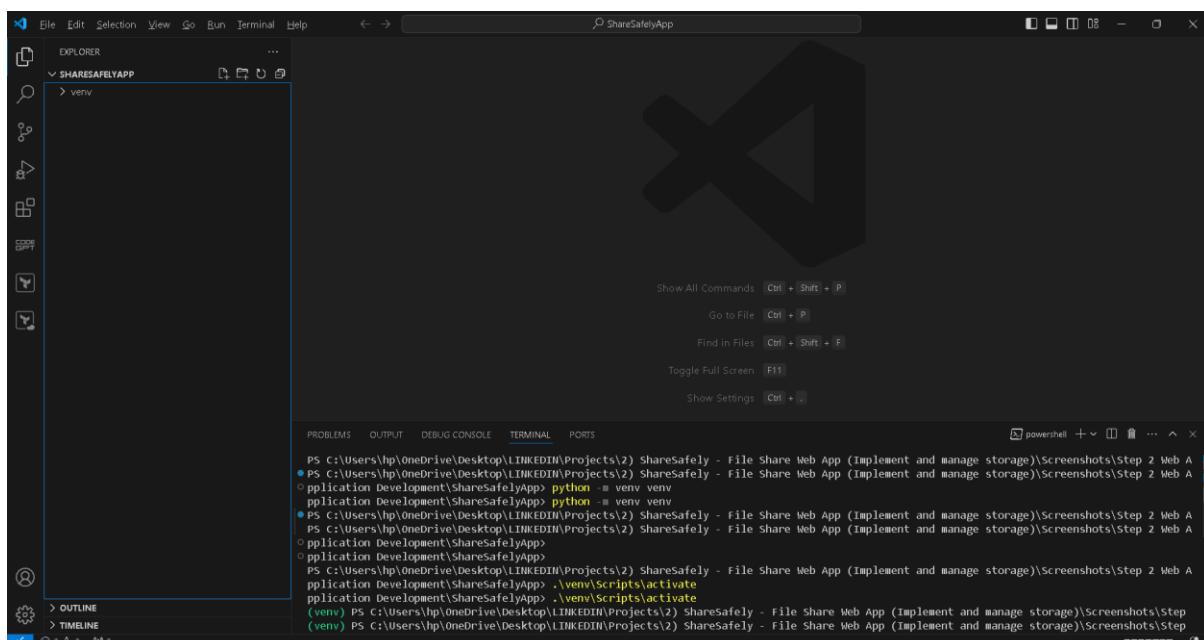
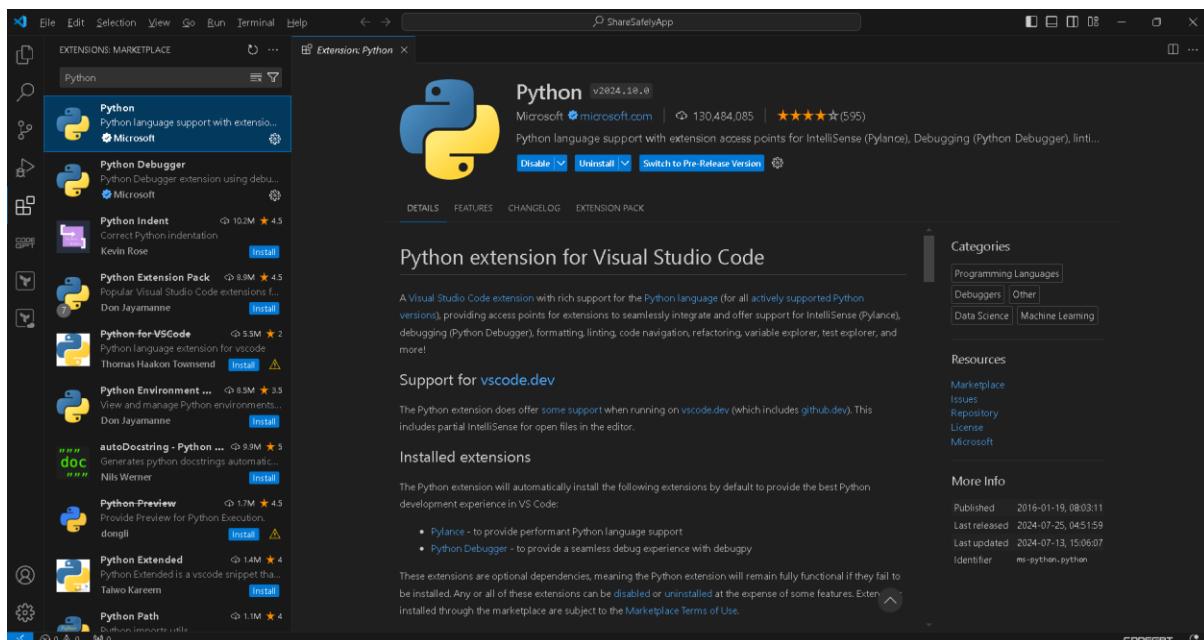
1. **File Upload and SAS Token Generation:**
  - **app.py:** Manages file uploads and generates SAS tokens.
2. **HTML Templates:**
  - **upload.html and link.html:** Interface for uploading files and displaying links.

## Local Testing

After developing the application, I tested it locally to ensure functionality and usability.

- 1. Command to Run Application:** python app.py
- 2. Testing Environment:** VS Code terminal
- 3. Results:** The web application performed as expected, allowing file uploads and generating time-limited links successfully.

## Screenshots:

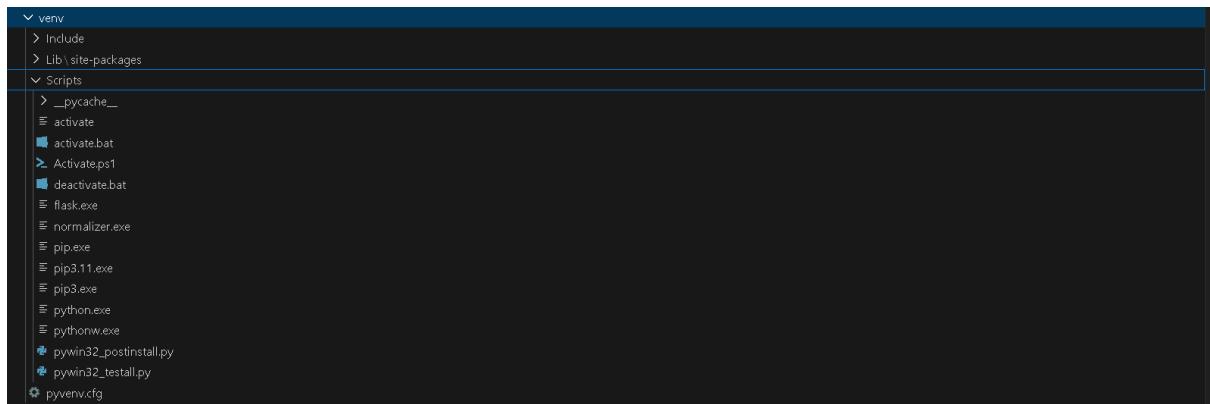


```
(venv) PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafe> pip install Flask azure-storage-blob azure-identity
Collecting Flask
  Downloading flask-3.0.3-py3-none-any.whl (101 kB)
Collecting azure-storage-blob
  Downloading azure_storage_blob-12.21.0-py3-none-any.whl (396 kB)
Collecting azure-identity
  Downloading azure_identity-1.17.1-py3-none-any.whl (173 kB)
Collecting Werkzeug>=3.0.0
  Downloading werkzeug-3.0.3-py3-none-any.whl (227 kB)
Collecting Jinja2>=3.1.2
  Downloading jinja2-3.1.4-py3-none-any.whl (133 kB)
Collecting itsdangerous>=2.1.2
  Downloading itsdangerous-2.2.0-py3-none-any.whl (16 kB)
Collecting click>=8.1.3
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
Collecting blinker>=1.6.2
  Downloading blinker-1.8.2-py3-none-any.whl (9.5 kB)
Collecting azure-core>=1.28.0
  Downloading azure_core-1.30.2-py3-none-any.whl (194 kB)
Collecting cryptography>=2.1.4
  Downloading cryptography-43.0.0-cp39-win_amd64.whl (3.1 MB)
Collecting typing_extensions>=4.6.0
  Downloading typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Collecting isodate>=0.6.1
  Downloading isodate-0.6.1-py2.py3-none-any.whl (41 kB)
Collecting msal>=1.24.0
  Downloading msal-1.30.0-py3-none-any.whl (111 kB)
Collecting msal_extensions>=0.3.0
  Downloading msal_extensions-1.2.0-py3-none-any.whl (19 kB)
collecting requests>=2.1.0
```

```
(venv) PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafe> pip install Flask azure-storage-blob azure-identity
Collecting msal-extensions>=0.3.0
  Downloading msal_extensions-1.7.0-py3-none-any.whl (19 kB)
Collecting requests>=2.21.0
  Downloading requests-2.32.3-py3-none-any.whl (64 kB)
Collecting six>=1.11.0
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting colorama
  Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Collecting cffi>=1.12
  Downloading cffi-1.16.0-cp311-cp311-win_amd64.whl (181 kB)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-2.1.5-cp311-cp311-win_amd64.whl (17 kB)
Collecting PyJWT<(3,>=1.0.0
  Downloading PyJWT-2.8.0-py3-none-any.whl (22 kB)
Collecting portalocker>=1.4
  Downloading portalocker-2.10.1-py3-none-any.whl (18 kB)
Collecting pycparser
  Downloading pycparser-2.22-py3-none-any.whl (117 kB)
Collecting pywin32>=226
  Using cached pywin32-306-cp311-cp311-win_amd64.whl (9.2 MB)
Collecting charset_normalizer>=4,>=2
  Downloading charset_normalizer-3.3.2-cp311-cp311-win_amd64.whl (99 kB)
Collecting idna<4,>=2.5
  Downloading idna-3.7-py3-none-any.whl (66 kB)
Collecting urllib3<3,>=1.21.1
  Downloading urllib3-2.2.2-py3-none-any.whl (121 kB)
Collecting certifi>=2017.4.17
  Downloading certifi-2024.7.4-py3-none-any.whl (162 kB)
Installing collected packages: pywin32, urllib3, typing_extensions, six, PyJWT, pycparser, portalocker, MarkupSafe, itsdangerous, idna, colorama, charset_normalizer, certifi, blinker, Werkzeug, requests, Jinja2, isodate, click, cffi, Flask, cryptography, azure-core, azure-storage-blob, msal, msal_extensions, azure-identity
Successfully installed Flask-3.0.3 Jinja2-3.1.4 MarkupSafe-2.1.5 PyJWT-2.8.0 Werkzeug-3.0.3 azure-core-1.30.2 azure-identity-1.17.1 azure-storage-blob-12.21.0 blinker-7.4 cffi-1.16.0 charset-normalizer-3.3.2 click-8.1.7 colorama-0.4.6 cryptography-43.0.0 idna-3.7
```

The screenshot shows the Visual Studio Code interface with the following details:

- Terminal Tab:** The terminal window displays the output of a pip install command for a project named "ShareSafetyApp". The output includes the download of various Python packages and their dependencies, such as charset-normalizer, idna, urllib3, certifi, and flask.
- Explorer Tab:** The file explorer sidebar shows a folder named "SHARESAFELYAPP" containing a "venv" folder.
- Bottom Status Bar:** The status bar indicates the current file is "CODEGPT.ipynb" and shows other standard status icons.



```
File Edit Selection View Go Run Terminal Help ...
```

SHARESAFEYAPP

- static
- # styles.css
- templates
- link.html
- uploadhtml
- venv
- app.py

CODE GPT

OUTLINE

TIMELINE

Ln 36 Col 64 Spaces: 4 UFT-8 CR/LF Python 3.11.1 (venv/ venv) CODEGPT

```
File Edit Selection View Go Run Terminal Help ...
```

SHARESAFEYAPP

app.py > Upload file

```
1 from flask import Flask, request, redirect, url_for, render_template
2 from azure.storage.blob import BlobServiceClient, generate_blob_sas, BlobSasPermissions
3 from datetime import datetime, timedelta
4
5 app = Flask(__name__)
6
7 # Configuration
8 STORAGE_ACCOUNT_NAME = 'sharesafelystrg'
9 CONTAINER_NAME = 'uploads'
10 ACCOUNT_KEY = [REDACTED]
11
12 # Construct the connection string
13 connection_string = (
14     "DefaultEndpointsProtocol=https;"
15     f"AccountName={STORAGE_ACCOUNT_NAME};"
16     f"AccountKey={ACCOUNT_KEY};"
17     "EndpointSuffix=core.windows.net"
18 )
19
20 # Initialize BlobServiceClient with the connection string
21 blob_service_client = BlobServiceClient.from_connection_string(connection_string)
22
23 @app.route('/', methods=['GET', 'POST'])
24 def upload_file():
25     if request.method == 'POST':
26         file = request.files.get('file')
27         if file and file.filename:
28             blob_client = blob_service_client.get_blob_client(container=CONTAINER_NAME, blob=file.filename)
29             blob_client.upload_blob(file, overwrite=True)
30             sas_token = generate_blob_sas(
31                 account_name=STORAGE_ACCOUNT_NAME,
32                 container_name=CONTAINER_NAME,
33                 blob_name=file.filename,
34                 account_key=ACCOUNT_KEY,
35                 permission=BlobSasPermissions(read=True),
36                 expiry=datetime.utcnow() + timedelta(seconds=15)
37             )
```

CODE GPT

OUTLINE

TIMELINE

Ln 37 Col 1 Spaces: 4 UFT-8 CR/LF Python 3.11.1 (venv/ venv) CODEGPT

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure for "SHARESAFEYAPP" with files like "static", "templates", "venv", and "app.py".
- Code Editor:** Displays the "app.py" file content:

```
def upload_file():
    blob_name=file.filename,
    account key=ACCOUNT KEY,
    permission=BlobSasPermissions(read=True),
    expiry=datetime.utcnow() + timedelta(seconds=15)
    sas_url = f"https://{STORAGE ACCOUNT NAME}.blob.core.windows.net/{CONTAINER NAME}/{file.filename}?{sas_token}"
    return redirect(url_for('file_link', url=sas_url))
return render_template('upload.html')

@app.route('/link')
def file_link():
    url = request.args.get('url')
    return render_template('link.html', url=url)

if __name__ == '__main__':
    app.run(debug=True)
```
- Status Bar:** Shows "Ln 37, Col 1" and "CODEGPT".

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure for "SHARESAFEYAPP" with files like "static", "templates", "venv", and "app.py".
- Code Editor:** Displays the "upload.html" file content:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{{ url_for('static', filename='styles.css') }}">
    <title>Upload File - ShareSafeLY</title>
</head>
<body>
    <div class="container">
        <h1>ShareSafeLY File Upload</h1>
        <form action="/" method="post" enctype="multipart/form-data">
            <label for="file">Choose a file to upload:</label><br>
            <input type="file" name="file" id="file" required><br>
            <button type="submit">Upload</button>
        </form>
    </div>
</body>
</html>
```
- Status Bar:** Shows "Ln 20, Col 1" and "CODEGPT".

The screenshot shows the VS Code interface with the title bar "ShareSafetyApp". The Explorer sidebar on the left shows a project structure with a folder "SHARESAFEAPP" containing "static", "templates", "venv", and "app.py". The "templates" folder contains "link.html", "uploadhtml", and "uploadcss". The "link.html" file is selected and open in the main editor area. The code in "link.html" is:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{{ url_for('static', filename='styles.css') }}>
    <title>File Uploaded - ShareSafety</title>
</head>
<body>
    <div class="container">
        <h1>File Uploaded Successfully!</h1>
        <p>Share this link to provide access to the file:</p>
        <a href="{{ url }}" target="_blank">{{ url }}</a>
    </div>
</body>
</html>
```

The status bar at the bottom indicates "Ln 17 Col 1 Spaces:4 UTF-8 CR/LF HTML CODEGPT".

The screenshot shows the VS Code interface with the title bar "ShareSafetyApp". The Explorer sidebar on the left shows a project structure with a folder "SHARESAFEAPP" containing "static", "templates", "venv", and "app.py". The "static" folder contains "styles.css". The "styles.css" file is selected and open in the main editor area. The code in "styles.css" is:

```
body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    color: #333;
    margin: 0;
    padding: 0;
}

.container {
    max-width: 600px;
    margin: 50px auto;
    padding: 20px;
    background-color: #fff;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    border-radius: 5px;
}

h1 {
    color: #007bff;
    text-align: center;
}

form {
    display: flex;
    flex-direction: column;
    align-items: center;
}

label, input, button {
    width: 100%;
    margin: 10px 0;
}

button {
    padding: 10px;
    background-color: #007bff;
    border: none;
}
```

The status bar at the bottom indicates "Ln 57 Col 1 Spaces:4 UTF-8 CR/LF CSS CODEGPT".

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "SHARESAFEAPP".
- Editor:** Displays the content of the "styles.css" file.
- Bottom Status Bar:** Shows "Ln 57, Col 1" and other file-related information.

```
static > # styles.css > ...  
32 }  
33 button {  
34   padding: 10px;  
35   background-color: #007BFF;  
36   border: none;  
37   border-radius: 5px;  
38   color: white;  
39   cursor: pointer;  
40   font-size: 16px;  
41 }  
42 button:hover {  
43   background-color: #0056b3;  
44 }  
45 a {  
46   color: #007BFF;  
47   text-decoration: none;  
48   word-break: break-all;  
49 }  
50 a:hover {  
51   text-decoration: underline;  
52 }  
53  
54  
55  
56  
57
```

---

## Secure Credentials with Azure Key Vault

In this section, I will document the steps I took to secure the credentials for my "ShareSafely - File Share Web App" using Azure Key Vault. This process ensures that sensitive information, such as the storage account key, is not hard-coded into the application but securely stored and accessed.

### Introduction

To enhance security and prevent the exposure of sensitive credentials in my application code, I used Azure Key Vault. This service securely stores secrets like the storage account key, which my application needs to access Azure Blob Storage.

### Installing Required Packages

First, I installed the necessary Azure SDKs to interact with Azure Key Vault and manage secrets. The command used is:

```
pip install azure-identity azure-keyvault-secrets
```

## **Creating an Azure Key Vault**

Next, I created a Key Vault in the Azure portal to securely store the storage account key.

### 1. Key Vault Creation:

- Service: Azure Key Vault
- Key Vault Name: myKeyVaultShareSafely
- Purpose: To store secrets securely.

## **Managing Access to Key Vault**

To manage the secrets in Azure Key Vault, I assigned myself the "Key Vault Secrets Officer" role. This role allows me to add, delete, and manage secrets within the Key Vault.

### 1. Role Assignment:

- Role: Key Vault Secrets Officer
- Assignee: Myself (Vivek Vashisht)
- Purpose: To manage the secrets stored in the Key Vault.

## **Adding a Secret to Key Vault**

With the Key Vault set up and access permissions configured, I added the storage account key as a secret.

### 1. Adding the Secret:

- Secret Name: blob-storage-account-key
- Value: The storage account key from Azure Blob Storage
- Steps:
  - In the Key Vault, navigate to the "Secrets" section.
  - Click "Generate/Import" to create a new secret.
  - Enter the secret name and value.
  - Click "Create" to store the secret securely.

## **Updating the Application Code**

To securely access the storage account key stored in Azure Key Vault, I updated my application's code (app.py). The updated code retrieves the key using Azure SDKs, ensuring that the key is not exposed in the code.

## **Testing and Verification**

I tested the application to ensure that it correctly retrieves the storage account key from Azure Key Vault without exposing it in the code.

### 1. Verification:

- The application successfully accessed the storage account key from Azure Key Vault.
- The key was not visible in the application code or configuration files.

## 2. Security Check:

- Ensured that all access controls and security measures were correctly implemented.
- 

### Screenshots:

```
PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafeApp
pip install azure-identity
Requirement already satisfied: azure-identity in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (1.17.1)
Collecting azure-keyvault-secrets
  Downloading azure_keyvault_secrets-4.8.0-py3-none-any.whl (82 kB)
Requirement already satisfied: azure-core>=1.23.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (0.6.1)
Requirement already satisfied: cryptography<2.5 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (2.3.2)
Requirement already satisfied: msal>=1.24.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (1.30.0)
Requirement already satisfied: msal-extensions>=0.3.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (1.3.0)
Requirement already satisfied: typing-extensions>=4.0.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (4.12.2)
Requirement already satisfied: isodate>=0.6.1 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (0.6.1)
Requirement already satisfied: requests>=2.21.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (2.23.0)
Requirement already satisfied: six>=1.11.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (1.16.0)
Requirement already satisfied: cftrio>=1.12 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from cryptography>=2.5>azure-identity) (1.16.0)
Requirement already satisfied: PyMOTIF[crypto]>=1.0.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from msal>=1.24.0>azure-identity) (2.8.0)
Requirement already satisfied: portalocker>=1.4 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from msal-extensions>=0.3.0>azure-identity) (2.10.1)
Requirement already satisfied: pycparser in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from msal>=1.24.0>azure-identity) (2.32.3)
Requirement already satisfied: pywin32>=226 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from cryptography>=2.5>azure-identity) (2.22)
Requirement already satisfied: Requirement already satisfied: portalocker>=1.4 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from portalocker>=1.4>msal-extensions>=0.3.0>azure-identity) (3.06)
Requirement already satisfied: charset-normalizer>=2 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from requests>=2.21.0>azure-core>=1.23.0>azure-identity) (3.3.2)
Requirement already satisfied: idna[<2.5 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-keyvault-secrets) (0.6.1)
Requirement already satisfied: requests>=2.21.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from azure-core>=1.23.0>azure-identity) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\hp\onedrive\desktop\linkedin\projects\2) sharessafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\sharessafelyapp\venv\lib\site-packages (from requests>=2.21.0>azure-core>=1.23.0>azure-identity) (2.2.2)
```

```
PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafeApp
[notice] A new release of pip available: 22.3.1 > 24.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip
○ PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafeApp
```

Key vaults - Microsoft Azure

portal.azure.com/?feature.msaljs=true#browse/Microsoft.KeyVault%2FVaults

Microsoft Azure

Key vaults

Default Directory (vivekvash1507@gmail.com)

+ Create Manage deleted vaults Manage view Re

Filter for any field... Subscription equals all Resource g

Showing 0 to 0 of 0 records.

Name ↑

All Services (10) Marketplace (31) More (4)

Services

- Key vaults
- SSH keys
- Microsoft Entra authentication methods
- BitLocker Keys

Marketplace

- Key Vault
- Language service
- Document Intelligence (form recognizer)
- SSH Key

Documentation

- What is Azure Key Vault?
- Overview of Key Management in Azure
- Monitoring Azure Key Vault

Continue searching in Microsoft Entra ID

Searching all subscriptions. Give feedback

No grouping List view

Subscription ↑↓ Tags

Give feedback

<https://portal.azure.com/?feature.msaljs=true#blade/HubExtension/BrowseResourceBlade/r...>

Create a key vault - Microsoft Azure

portal.azure.com/?feature.msaljs=true#create/Microsoft.KeyVault

Microsoft Azure

Home > Key vaults

Create a key vault

Review + create

View Automation Template

Basics

Subscription	Azure subscription 1
Resource group	ShareSafetyRG
Key vault name	myKeyVaultShareSafety
Region	East US
Pricing tier	Standard
Soft-delete	Enabled
Purge protection during retention period	Disabled
Days to retain deleted vaults	90 days

Access configuration

Azure Virtual Machines for deployment	Disabled
Azure Resource Manager for template deployment	Disabled
Azure Disk Encryption for volume encryption	Disabled

Previous Next Create Give feedback

Create a secret - Microsoft Azure

portal.azure.com/?feature.msaljs=true#view/Microsoft\_Azure\_KeyVault/CreateSecretBlade/secret-/hull/vaultId%2fsub... Home myKeyVaultShareSafely Secrets Create a secret

Upload options: Manual

Name \*: blob-storage-account-key

Secret value \*:

Content type (optional):

Set activation date:

Set expiration date:

Enabled: Yes

Tags: 0 tags

Create Cancel

Add role assignment - Microsoft Azure

portal.azure.com/?feature.msaljs=true#view/Microsoft\_Azure\_AD/AddRoleAssignmentsLandingBlade/scope/%2fsubscription... Home Subscriptions Azure subscription 1 | Access control (IAM) Add role assignment

Role: Key Vault Secrets Officer

Scope: /subscriptions/ee9ea131-d6f1-4e0b-bae-b293615685ae

Members:

Name	Object ID	Type
Vivek Vashist	8ad0491b-df2f-4e46-a1cf-5e9bc4d59471	User

Description: No description

Review + assign Previous Next Feedback

---

## **Deployment to Azure Web Apps**

In this section, I document the steps taken to deploy the "ShareSafely" web application to Azure Web Apps. This process involves preparing the application for deployment, setting up the Azure environment, and configuring necessary services for secure and efficient operation.

### **Preparing the Application for Deployment**

Before deploying the application to Azure, I needed to ensure all dependencies were listed in a requirements.txt file. This file is crucial for the Azure environment to understand what packages need to be installed for the application to run.

Generating requirements.txt: To generate the requirements.txt, I used the following command in the VS Code terminal:

```
pip freeze > requirements.txt
```

Contents of requirements.txt:

```
plaintext
Copy code
azure-core==1.30.2
azure-functions==1.20.0
azure-identity==1.17.1
azure-keyvault-secrets==4.8.0
azure-storage-blob==12.21.0
blinker==1.8.2
certifi==2024.7.4
cffi==1.16.0
charset-normalizer==3.3.2
click==8.1.7
colorama==0.4.6
cryptography==43.0.0
Flask==3.0.3
idna==3.7
```

```
isodate==0.6.1
itsdangerous==2.2.0
Jinja2==3.1.4
MarkupSafe==2.1.5
msal==1.30.0
msal-extensions==1.2.0
portalocker==2.10.1
pycparser==2.22
PyJWT==2.8.0
requests==2.32.3
six==1.16.0
typing_extensions==4.12.2
urllib3==2.2.2
Werkzeug==3.0.3
```

## **Creating and Configuring the Web App in Azure**

Creating the Web App:

1. I navigated to the Azure Portal and created a new Web App:
  - o Subscription: Selected my Azure subscription.
  - o Resource Group: Chose an existing resource group (ShareSafelyRG).
  - o Name: Entered "ShareSafelyApp".
  - o Publish: Selected "Code".
  - o Runtime Stack: Chose Python version (Python 3.12).
  - o Region: Selected the region East US.
  - o Linux Plan: Selected a plan (Free).

## **Deploying the Application**

Installing Azure App Service Extension: I installed the Azure App Service Extension in Visual Studio Code to facilitate the deployment process.

Deployment Steps:

1. Sign in to Azure Account: I used the Azure App Service extension to sign in to my Azure account.
2. Deploying the Application:
  - o From the resources, I expanded my Azure Subscription, then expanded the App Services.
  - o I right-clicked on "ShareSafelyApp" and clicked "Deploy to Web App...".
  - o I chose the project folder (ShareSafelyApp) where I developed the web application in VS Code.
  - o Confirmed the deployment when prompted.

## **Configuring Environment Variables**

Setting Environment Variables: In the Azure Portal, for the deployed Web App, I set the following environment variables:

- KEY\_VAULT\_URI: URI of the Azure Key Vault.

- SECRET\_NAME: Name of the secret in the Key Vault containing the storage account key.

## Enabling Managed Identity and Configuring Access

### Enabling Managed Identity:

1. Accessed the "Identity" section under "Settings" in my Web App.
2. Enabled "System Assigned Identity" by setting "Status" to "On" and saved changes. This assigns a managed identity to my web app.

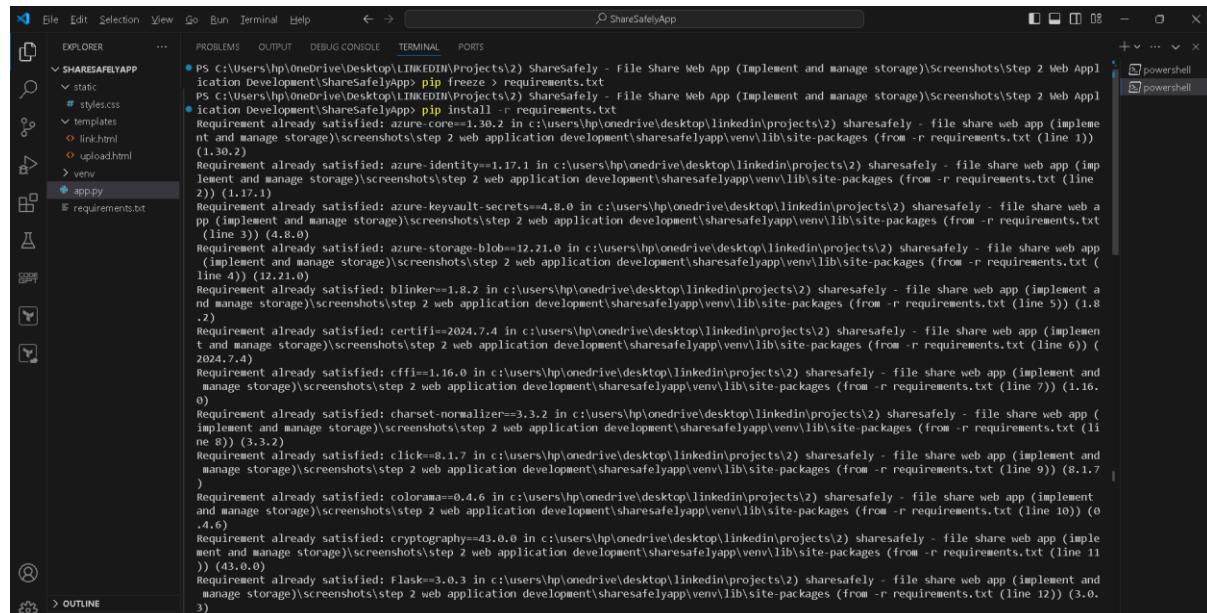
### Assigning Key Vault Access:

1. Navigated to my Azure Key Vault in the Azure Portal.
2. Went to "Access Control (IAM)".
3. Added a Role Assignment of "Key Vault Secrets Officer" to the managed identity of the Azure Web App (ShareSafelyApp).
4. Restarted the Azure Web App in the Azure Portal.

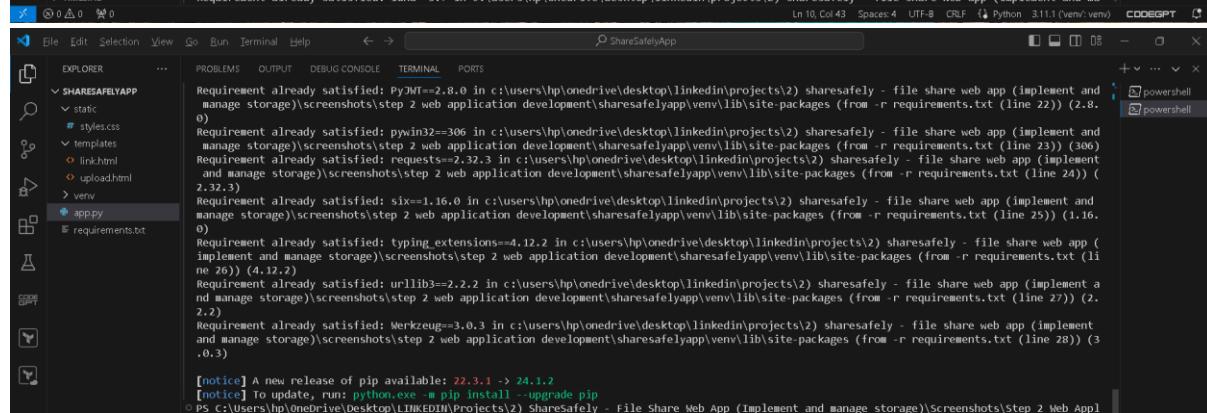
## Verification and Final Testing

After deploying and configuring the application, I tested it to ensure it was working correctly in the Azure Web App environment. The web application functioned as expected, demonstrating successful deployment and configuration.

## Screenshots:



```
PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\screenshots\Step 2 Web Application Development\ShareSafelyApp> pip freeze > requirements.txt
PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\screenshots\Step 2 Web Application Development\ShareSafelyApp> pip install -r requirements.txt
Requirement already satisfied: azure-core==1.30.2 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 1)) (1.30.2)
Requirement already satisfied: azure-identity==1.17.1 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 2)) (1.17.1)
Requirement already satisfied: azure-keyvault-secrets==4.8.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 3)) (4.8.0)
Requirement already satisfied: azure-storage-blob==12.21.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 4)) (12.21.0)
Requirement already satisfied: blinker==1.8.2 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 5)) (1.8.2)
Requirement already satisfied: certifi==2024.7.4 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 6)) (2024.7.4)
Requirement already satisfied: cffi==1.16.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 7)) (1.16.0)
Requirement already satisfied: charset-normalizer==3.3.2 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 8)) (3.3.2)
Requirement already satisfied: click==8.1.7 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 9)) (8.1.7)
Requirement already satisfied: colorama==0.4.6 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 10)) (0.4.6)
Requirement already satisfied: cryptography==43.0.0 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 11)) (43.0.0)
Requirement already satisfied: Flask==3.0.3 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and manage storage)\screenshots\step 2 web application development\shareSafelyApp\venv\lib\site-packages (from -r requirements.txt (line 12)) (3.0.3)
Requirement already satisfied: idna==3.7 in c:\users\hp\onedrive\desktop\linkedin\projects\2) shareSafely - file share web app (implement and ma
[notice] A new release of pip available: 22.3.1 -> 24.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip
PS C:\Users\hp\OneDrive\Desktop\LINKEDIN\Projects\2) ShareSafely - File Share Web App (Implement and manage storage)\Screenshots\Step 2 Web Application Development\ShareSafelyApp>
```



The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Standard window controls (Minimize, Maximize, Close).
- Explorer:** Shows a project structure for "SHARESAFEAPP" with folders like static, templates, and a file named "app.py".
- Code Editor:** The "requirements.txt" file is open, displaying a list of Python dependencies:

```
1 azure-core==1.30.2
2 azure-identity==1.17.1
3 azure-keyvault-secrets==4.8.0
4 azure-storage-blob==12.21.0
5 blinker==1.8.2
6 certifi==2024.7.4
7 cffi==1.16.0
8 charset-normalizer==3.3.2
9 click==8.1.7
10 colorama==0.4.6
11 cryptography==43.0.0
12 flask==3.0.3
13 idna==3.7
14 isodate==0.6.1
15 itsdangerous==2.2.0
16 Jinja2==3.1.4
17 MarkupSafe==2.1.5
18 msal==1.30.0
19 msal-extentions==1.2.0
20 portalocker==2.10.1
21 pycparser==2.22
22 PyJWT==2.8.0
23 pywin32==306
24 requests==2.32.3
25 six==1.16.0
26 typing_extensions==4.12.2
27 urllib3==2.2.2
28 Werkzeug==3.0.3
29 |
```

- Bottom Status Bar:** Ln 29, Col 1, Spaces: 4, UTF-16 LE, CRLF, pip requirements, CODEGPT.

The screenshot shows the Microsoft Azure App Services dashboard with the following details:

- Header:** App Services - Microsoft Azure, portal.azure.com/?feature.msaljs=true#browse/Microsoft.Web%2Fsites, Microsoft Azure, Search resources, services, and docs (G+).
- Left Sidebar:** Home > App Services > ...
- Main Content:** A search bar and filter options for managing apps. The filters are set to "Subscription equals all", "Resource group equals all", and "Location equals all".
  - Buttons: Create, Manage Deleted Apps, Manage view, Refresh, Export to CSV, Open query, Assign tags, Start, Restart, Stop, Delete.
  - Search: Search resources, services, and docs (G+).
  - Filter: No grouping, List view.
  - Ordering: Status ↑↓, Location ↑↓, Pricing Tier ↑↓, App Service Plan ↑↓, Subscription ↑↓, App Type ↑↓.
- Center:** A large circular icon with a gear-like pattern. Below it, the text "No app services to display" is displayed.
- Bottom:** A call-to-action button "Learn more about App Service" and a feedback link "Give feedback".

Create Web App - Microsoft Azure

portal.azure.com/?feature.msaljs=true#create/Microsoft.WebSite

Microsoft Azure

Search resources, services, and docs (G+)

vivekvash1507@gmail.com

DEFAULT DIRECTORY (vivekvash1507@gmail.com)

Home > App Services >

Create Web App

Basics Deployment Networking Monitor + secure Tags Review + create

Summary

Web App by Microsoft

Free sku Estimated price - Free

Basic authentication for this app is currently disabled and may impact deployments. Click to learn more.

Details

Subscription: ShareSafetyRG

Resource Group: ShareSafetyApp

Name: ShareSafetyApp

Publish: Code

Runtime stack: Python 3.12

App Service Plan (New)

Name: ASP-ShareSafetyRG-91b3

Operating System: Linux

Create < Previous Next > Download a template for automation

File Edit Selection View Go Run Terminal Help

EXTENSIONS MARKETPLACE

Azure

Azure App Service An Azure App Service management extension for Visual Studio Code Microsoft

Azure Storage Manage your Azure Storage accounts including Blob, Queue, Table, and File Microsoft

Azure Databases Create, browse, and update globally distributed, multi-region databases Microsoft

Azure Account A common Sign In and Subscription management extension Microsoft

Azure Resources An extension for viewing and managing Azure resources Microsoft

Azure Functions An Azure Functions extension for Visual Studio Code Microsoft

Azure Repos Remotely browse and edit any Azure Repos Microsoft

Azure Machine Learning Visual Studio Code extension for Azure Machine Learning Microsoft

Azure Machine Learning - Remote This extension is used by the Azure Machine Learning extension Microsoft

Azure Virtual Machines An extension for managing Azure Virtual Machines Microsoft

Extension: Azure App Service

Azure App Service v0.25.2 Preview

Microsoft microsoft.com 2,041,993 4.5 (30)

An Azure App Service management extension for Visual Studio Code.

Disable Uninstall

DETAILS FEATURES CHANGELOG DEPENDENCIES

Azure App Service for Visual Studio Code (Preview)

Visit the [wiki](#) for more information about Azure App Service and how to use the advanced features of the extension.

Sign up today for your free Azure account and receive 12 months of free popular services, \$200 free credit and 25+ always free services. [Start Free](#)

Installation

- Download and install the [Azure App Service extension for Visual Studio Code](#)
- If you're interested in deploying single page web apps or progressive web apps (something **without** an express server), install the [Azure Storage extension](#)
- Wait for the extension to finish installing then reload Visual Studio Code when prompted
- Once complete, you'll see an Azure icon in the Activity Bar

If your activity bar is hidden, you won't be able to access the extension. Show the Activity Bar by clicking [View > Appearance > Show Activity Bar](#)

- Sign in to your Azure Account by clicking [Sign in to Azure...](#)

Categories

Azure

Resources

Marketplace Issues Repository License Microsoft

More Info

Published 2017-09-19, 11:28:18

Last released 2024-02-29, 13:45:06

Identifier ms-azurertools.vscode-azureappservice

CODEGPT

The screenshot shows the Azure DevOps interface with the 'ShareSafetyApp' deployment configuration open. The 'Deploy to Web App...' context menu is selected, showing options like Start, Stop, Restart, Delete, and a 'Deploy' button. The code editor displays Python code for initializing a BlobServiceClient with a connection string from a Key Vault secret.

```
from flask import Flask, request, redirect, url_for, render_template
from azure.storage.blob import BlobServiceClient, generate_blob_sas, BlobSasPermissions
from azure.identity import DefaultAzureCredential
from azure.keyvault.secrets import SecretClient
from datetime import timedelta

app = Flask(__name__)

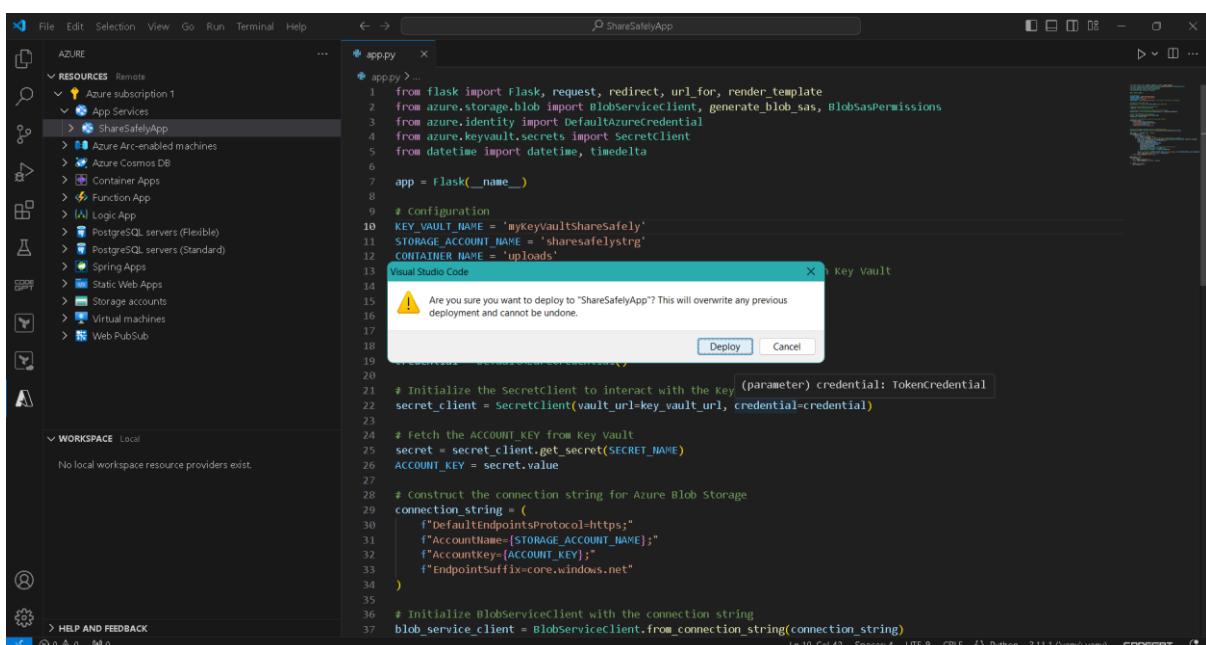
# Configuration
KEY_VAULT_NAME = 'mykeyVaultShareSafety'
STORAGE_ACCOUNT_NAME = 'sharesafelystrg'
CONTAINER_NAME = 'uploads'

# Initialize the SecretClient to interact with the Key Vault (parameter) credential: TokenCredential
secret_client = SecretClient(vault_url=key_vault_url, credential=credential)

# Fetch the ACCOUNT_KEY from key vault
secret = secret_client.get_secret(SECRET_NAME)
ACCOUNT_KEY = secret.value

# Construct the connection string for Azure Blob Storage
connection_string = (
    f'DefaultEndpointsProtocol=https;'
    f'AccountName={STORAGE_ACCOUNT_NAME};'
    f'AccountKey={ACCOUNT_KEY};'
    f'EndpointSuffix=core.windows.net'
)

# Initialize BlobServiceClient with the connection string
blob_service_client = BlobServiceClient.from_connection_string(connection_string)
```



```

    from flask import Flask, request, redirect, url_for, render_template
    from azure.storage.blob import BlobServiceClient, generate_blob_sas, BlobSasPermissions
    from azure.identity import DefaultAzureCredential
    from azure.keyvault.secrets import SecretClient
    from datetime import datetime, timedelta

```

Deploy to App "ShareSafetyApp" Succeeded

Role	Members	Conditions	Review + assign
Key Vault Secrets Officer			
Scope	/subscriptions/ee9ea131-d6f1-4e0b-bae0-b2936156b5ae/resourceGroups/ShareSafetyRG/providers/Microsoft.KeyVault/vaults/myKeyVaultShareSafety		
Members	Name	Object ID	Type
	ShareSafetyApp	ebeb7f0e-2fdd-4e5d-9796-386a1652413b	App Service
Description	No description		

---

## **Monitoring and Maintenance**

### **Monitoring and Maintenance**

In this section, I will document the steps I took to set up monitoring and maintenance for the "ShareSafely" web application using Azure Monitor. This ensures the application remains performant and any issues are promptly addressed.

#### **Accessing Azure Monitor**

After deploying the web application to Azure Web Apps, I accessed Azure Monitor to set up monitoring and alerting mechanisms.

1. Accessing Azure Monitor:
  - o I went to the Azure Portal.
  - o In the left-hand menu, I selected "Monitor" to access Azure Monitor.

#### **Setting Up Metrics Monitoring**

To keep track of the application's performance and health, I configured various metrics within Azure Monitor.

1. Navigating to Metrics:
  - o In the Monitor section, I clicked on "Metrics".
2. Selecting the Resource:
  - o I chose the resource type (App Services) and selected the specific Web App, "ShareSafelyApp," that I wanted to monitor.
3. Adding Metrics:
  - o I added several key metrics by clicking "Add metric" and selecting the following:
    - CPU Time
    - Requests
    - Response Time
    - Memory Working Set

These metrics help in understanding the application's resource utilization and performance.

#### **Configuring Alerts**

To be notified of any issues with the web application, I set up alert rules in Azure Monitor.

1. Navigating to Alerts:
  - o In the Monitor section, I clicked on "Alerts".

## 2. Creating Alert Rules:

- I clicked "New Alert Rule" and selected the specific Web App "ShareSafelyApp" for monitoring.

## 3. Setting the Signal Name:

- I chose "Http Server Errors" as the signal to trigger the alert.

## 4. Defining the Alert Conditions:

- Threshold: Static
- Aggregation Type: Total
- Operator: Greater than
- Unit: Count
- Threshold Value: 1
- Lookback period: 5 minutes
- Check every: 1 minute

## 5. Setting Up an Action Group:

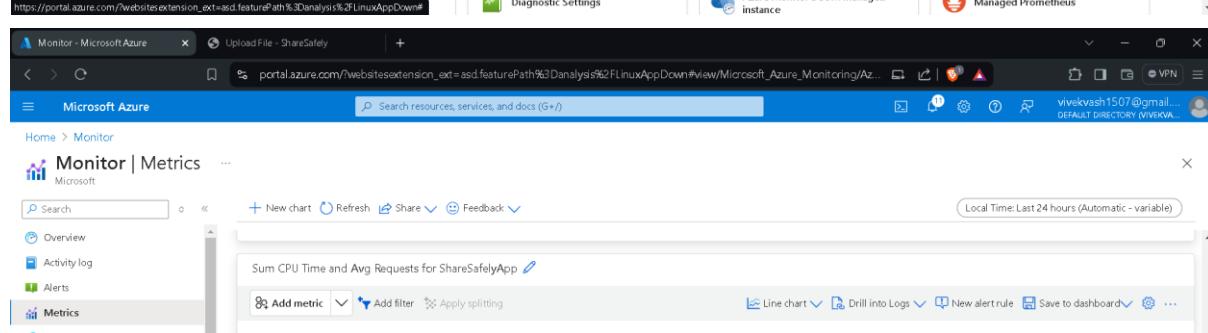
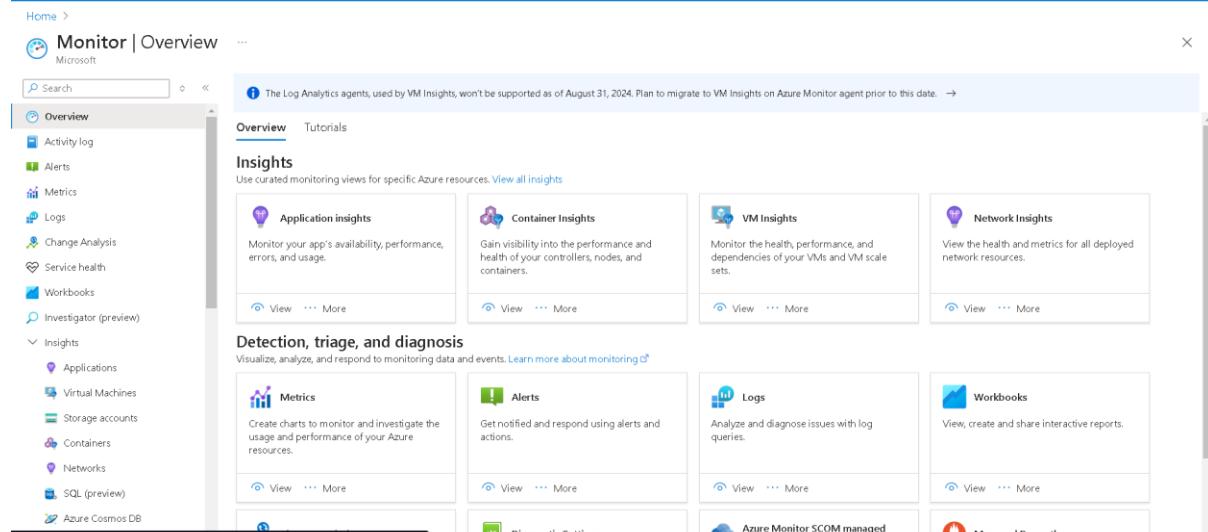
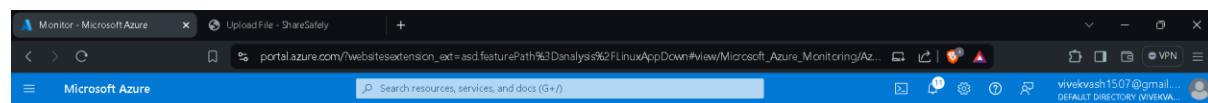
- I set up an action group to notify me via email in case of alerts.
- Action group name: HttpServerError
- Display name: HttpError

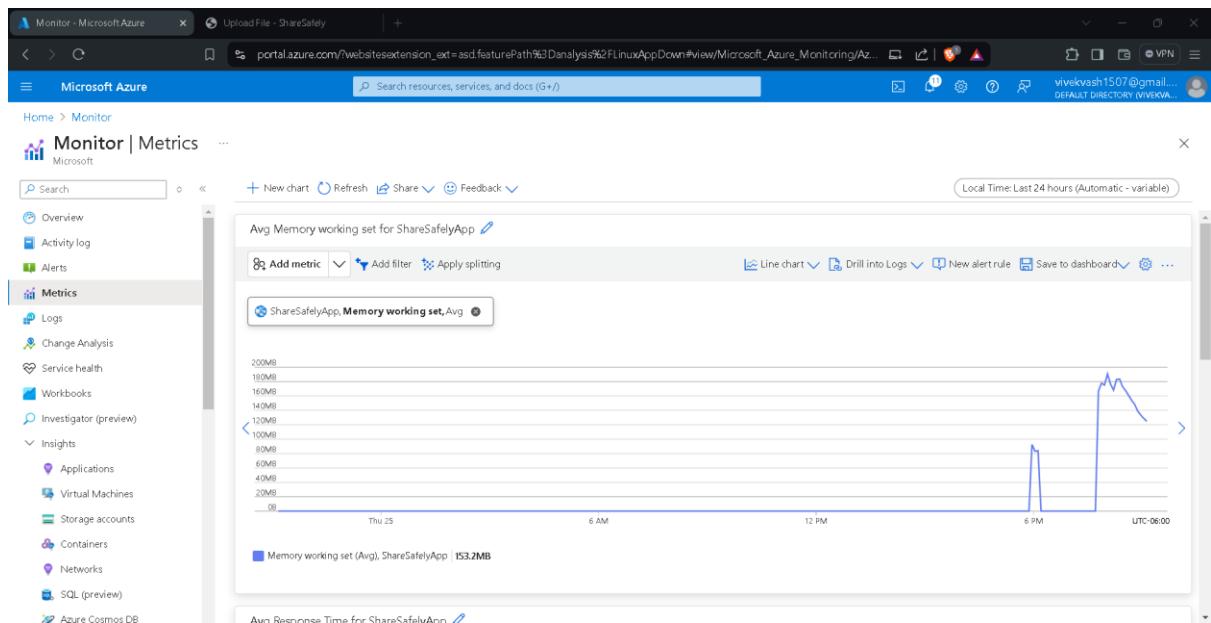
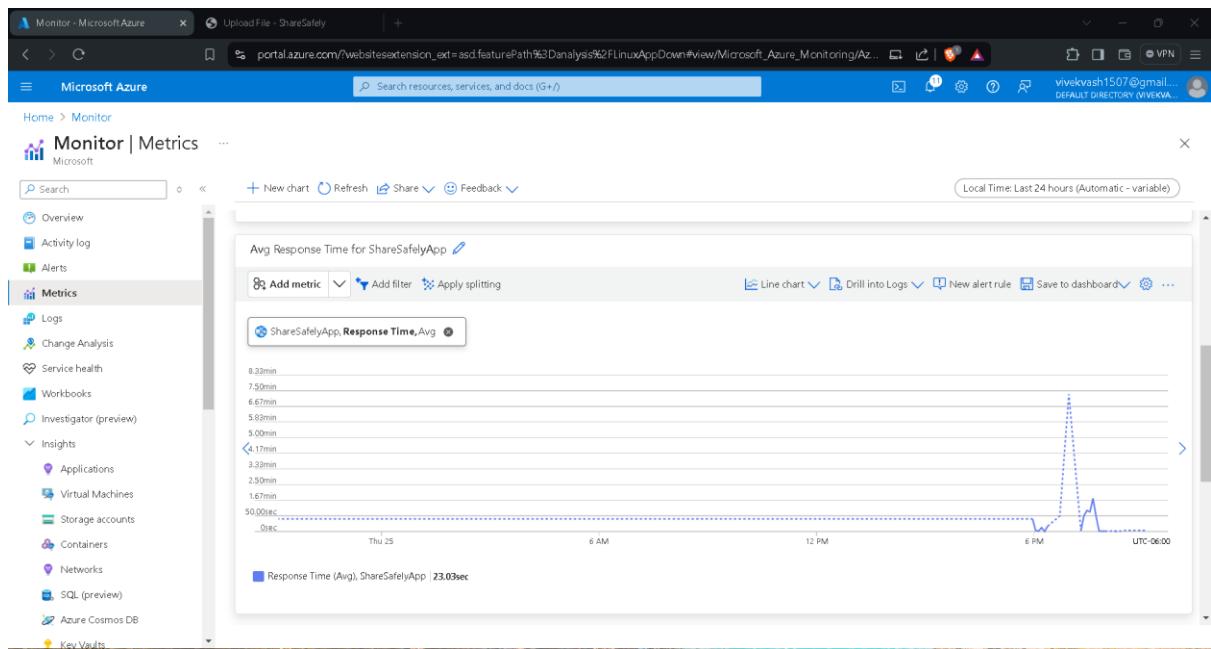
## 6. Actions:

- Email: I added my email address (vivekvash1507@gmail.com) to receive notifications.

After setting up the alert, I received an email confirming, "You're now in the HttpNotFound action group," indicating that the alert configuration was successful.

## Screenshots:





Screenshot of the Microsoft Azure Monitor Alerts page.

The page shows a summary of alerts:

- Total alerts: 0
- Critical: 0
- Error: 0
- Warning: 0
- Informational: 0
- Verbose: 0

Alert status: No alerts found

Message: Try changing your search or choose a different scope level if you don't see what you're looking for.

Navigation pane on the left includes:

- Overview
- Activity log
- Alerts
- Metrics
- Logs
- Change Analysis
- Service health
- Workbooks
- Investigator (preview)
- Insights (Applications, Virtual Machines, Storage accounts, Containers, Networks, SQL (preview), Azure Cosmos DB, Key Vaults)

Header bar includes: Home, Monitor, Alerts, View as timeline (preview), Create, Alert rules, Action groups, Alert processing rules, Prometheus rule groups, Change user response, Actions, Refresh, and a user profile.

Screenshot of the Microsoft Azure Create an alert rule page.

The page shows the "Condition" tab selected.

Signal name: Http Server Errors

Alert logic:

We have set the condition configuration automatically based on popular settings for this metric. Please review and make changes as needed.

Threshold: Static (radio button selected)

Aggregation type: Total

Operator: Greater than

Unit: Count

Threshold value: 1

Preview:

Whenever the total Http Server Errors is greater than 1

Time range: Over the last 6 hours

Time series: Aggregate

A line chart showing the count of Http Server Errors over the last 6 hours. The Y-axis ranges from 0 to 4. The X-axis shows hourly intervals. Red dots indicate values above the threshold of 1, with several spikes reaching up to 4.

Use quick actions (preview) - Microsoft Azure

UploadFile - ShareSafety

portal.azure.com/websitesextension\_ext=asd.featurePath%23Danalys%2FLinuxAppDown#view/Microsoft\_Azure\_Monitoring

vivekvash1507@gmail.com

Microsoft Azure

Search resources, services, and docs (G+)

Use quick actions (preview)

Create an alert rule

Actions

Scope Condition Actions Details Tags Review + create

An action group is a set of actions that can be applied to an alert rule. [Learn more](#)

Select actions

Use quick actions (preview)  
Select one or more of the quick actions.

Use action groups  
Add an existing action group or create a new one.

None

Quick actions

Quick actions not configured yet

Manage quick actions

Details

Action group name \*

Display name \*

Actions

Email

Email Azure Resource Manager Role

Azure mobile app notification

Review + create Previous Next: Details > Save Cancel

Create an alert rule - Microsoft Azure

UploadFile - ShareSafety

portal.azure.com/websitesextension\_ext=asd.featurePath%23Danalys%2FLinuxAppDown#view/Microsoft\_Azure\_Monitoring

vivekvash1507@gmail.com

Microsoft Azure

Search resources, services, and docs (G+)

Create an alert rule

Review + create

Scope Condition Actions Details Tags

Product details

Metric alert rule

1 Condition

0.10 USD/month

Pricing

Scope

Resource

Azure subscription 1 > ShareSafetyRG > ShareSafetyApp

Condition

Signal name: Http5xx  
Operator: Greater than  
Aggregation type: Total  
Threshold value: 1  
Lookback period: 5 minutes  
Check every: 1 minute

Create Previous

A Create an alert rule - Microsoft... | Upload File - ShareSafety

portal.azure.com/?websitesextension\_ext=asd.featurePath%3Danalysis%2FLinuxAppDown#view/Microsoft\_Azure\_Monitorin...

Microsoft Azure

Search resources, services, and docs (G+)

vivekvash1507@gmail...

DEFAULT DIRECTORY (VIVEKVA...)

Home > Monitor | Alerts >

Create an alert rule ...

**Actions**

Quick actions

**Details**

Action group name: HttpServerError  
Action group display name: HttpError

**Actions**

Email: vivekvash1507@gmail.com

**Details**

**Project details**

Subscription	Azure subscription 1
Resource group	ShareSafetyRG
Region	global

**Alert rule details**

Alert rule name	HttpServerError
Alert rule description	
Severity	3 - Informational
Enable upon creation	<input checked="" type="checkbox"/>

**Create** **Previous**

Monitor - Microsoft Azure | Upload File - ShareSafety

mail.google.com/mail/u/3/#inbox/1MfcqzQVxQDMmCSLpsDxTWlhdqJtkTg

Gmail

Compose

Inbox 1,179

Starred

Snoozed

Sent

Drafts

More

Labels

Notes

You're now in the **HttpError** action group

Microsoft Azure <azure-noreply@micrsoft.com> to me 9:23 PM (0 minutes ago)

You've been added to an Azure Monitor action group

You are now in the **HttpError** action group and will receive notifications sent to the group.

[View details on Azure Monitor action groups >](#)

**Account information**

Subscription ID [REDACTED]

Resource group name: ShareSafetyRG

Action group name: HttpServerError

To receive notifications from their actions, enable them.

---

## Azure Functions for Cleanup

In this section, I document the process of setting up a periodic cleanup of expired files in Azure Blob Storage using Azure Functions. This ensures that storage costs are managed effectively by automatically removing old files.

### Setting Up Azure Functions for Cleanup

After setting up alerts in Azure Monitor, I needed to create an automated cleanup process for expired files in Azure Blob Storage. Azure Functions provided an ideal solution for this task.

### Installing Azure Functions Extension in VS Code

First, I installed the Azure Functions extension in Visual Studio Code to facilitate the creation and deployment of serverless functions.

### Creating a New Azure Function Project

To begin, I created a new project for the cleanup function:

1. Clicked on the Azure icon in the VS Code sidebar.
2. Clicked the "Create New Project" icon in the Azure Functions section.
3. Chose a folder named CleanupFunction for my project.
4. Selected the Python language.
5. Chose Python version 3.11.
6. Selected the Timer Trigger template for the function.
7. Named the function BlobCleanupFunction.
8. Set the schedule using a CRON expression 0 0 \* \* \* (every day at midnight).

The directory structure created in the CleanupFunction project was as follows:

```
CleanupFunction/
├── .venv/
└── .vscode/
├── .funcignore
└── .gitignore
```

```
|--- function_app.py  
|--- host.json  
|--- local.settings.json
```

## Writing the Cleanup Logic

In the CleanupFunction directory, I created a file named `__init__.py` to define the logic for deleting expired blobs.

## Defining Dependencies

To ensure all necessary packages were included, I generated a `requirements.txt` file by running the following command in the VS Code terminal:

```
pip freeze > requirements.txt
```

Contents of `requirements.txt`:

```
plaintext  
Copy code  
azure-core==1.30.2  
azure-functions==1.20.0  
azure-identity==1.17.1  
azure-storage-blob==12.21.0  
blinker==1.8.2  
certifi==2024.7.4  
cffi==1.16.0  
charset-normalizer==3.3.2  
click==8.1.7  
colorama==0.4.6  
cryptography==43.0.0  
Flask==3.0.3  
idna==3.7  
isodate==0.6.1  
itsdangerous==2.2.0  
Jinja2==3.1.4  
MarkupSafe==2.1.5  
msal==1.30.0  
msal-extensions==1.2.0  
portalocker==2.10.1  
pycparser==2.22  
PyJWT==2.8.0  
requests==2.32.3  
six==1.16.0  
typing_extensions==4.12.2  
urllib3==2.2.2  
Werkzeug==3.0.3
```

## Deploying the Function App

I deployed the function app to Azure:

1. Expanded my Azure subscription in VS Code, then expanded the Function App.
2. Right-clicked on BlobCleanupService and selected "Deploy to Function App...".
3. Chose the project folder (CleanupFunction) where I developed the function app code.
4. Confirmed the deployment when prompted.

The function app was successfully deployed to Azure.

## Configuring Environment Variables

To set environment variables in Azure:

1. Went to the Azure Portal and navigated to my Function App.
2. Under Configuration > Application settings, I added a new setting for AZURE\_STORAGE\_CONNECTION\_STRING with the Blob Storage connection string.

## Testing the Function App

To test the function app:

1. In the Azure Portal, I navigated to the BlobCleanupFunction.
2. Clicked on "Test/Run" and selected the "\_master (Host key)".
3. Hit "Run" to test the function.

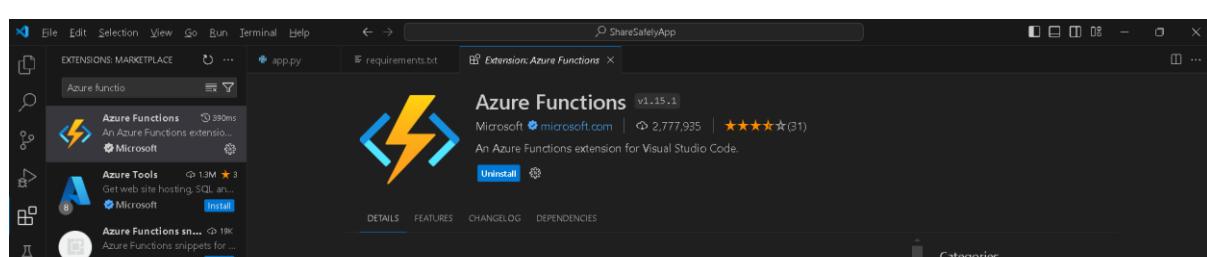
An error message related to Cross-Origin Resource Sharing (CORS) and network access restrictions appeared.

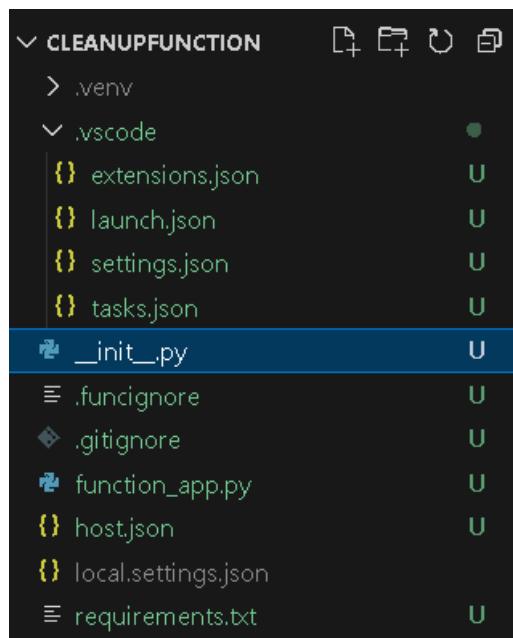
To resolve this:

- Configured CORS:
  1. Went to the Azure Portal.
  2. Navigated to my Function App.
  3. In the left-hand menu under "API", selected "CORS".
  4. Added "<https://portal.azure.com>" to the list of allowed origins.
  5. Clicked "Save".

After configuring CORS, the function worked perfectly as expected.

## Screenshots:





```
requirements.txt
1 azure-core==1.30.2
2 azure-functions==1.20.0
3 azure-identity==1.17.1
4 azure-storage-blob==12.21.0
5 blinker==1.8.2
6 certifi==2024.7.4
7 cffi==1.16.0
8 charset-normalizer==3.3.2
9 click==8.1.7
10 colorama==0.4.6
11 cryptography==43.0.0
12 Flask==3.0.3
13 idna==3.7
14 isodate==0.6.1
15 itsdangerous==2.2.0
16 Jinja2==3.1.4
17 MarkupSafe==2.1.5
18 msal==1.30.0
19 msal-extentions==1.2.0
20 portalocker==2.10.1
21 pyparser==2.22
22 PyJWT==2.8.0
23 requests==2.32.3
24 six==1.16.0
25 typing_extensions==4.12.2
26 urllib3==2.2.2
27 Werkzeug==3.0.3
28
```

The screenshot shows the Microsoft Azure portal interface for a BlobCleanupFunction. On the left, the function's code is displayed in a Python file named `function_app.py`:

```
1 import logging
2 import azure.functions as func
3
4 app = func.FunctionApp()
5
6 @app.schedule(schedule="0 0 * * *", arg_name="myTimer", run_on_startup=True,
7 ...            use_monitor=False)
8 def BlobCleanupFunction(myTimer: func.TimerRequest) -> None:
9     ... if myTimer.past_due:
10         ...     logging.info('The timer is past due!')
11
12     ... logging.info('Python timer trigger function executed')
```

The right side of the screen shows the "Test/Run" interface. It has tabs for "Input" and "Output". Under "Input", there is a dropdown menu set to "master (Host key)". The "Output" tab displays the results of a test run:

Provide parameters to test your function. Results can be found in the Output tab.

Key \*

Body

Run Close

The screenshot shows the Microsoft Azure portal interface for the same BlobCleanupFunction. The code remains the same as in the previous screenshot.

The "Test/Run" interface now shows the results of a successful test run:

HTTP response code

HTTP response content

Run Close

A BlobCleanupFunction - Microsoft Azure

portal.azure.com/?feature.msalsjs=true#view/Websites&extension/Function&blobMenuBlade=~/code/Test/resourceId/%2F... Update

Microsoft Azure

Search resources, services, and docs (G+)

Home > Resource groups > ShareSafely-Cleanup > BlobCleanupService >

BlobCleanupFunction | Code + Test

BlobCleanupService

Code + Test Integration Function Keys Invocations Logs Metrics

Save Discard Refresh TestRun Get function URL Disable Delete Upload Resource JSON Send us your feedback

This function has been edited through an external editor. Portal editing is disabled.

BlobCleanupService / function\_app.py

```
1 import logging
2 import azure.functions as func
3
4 app = func.FunctionApp()
5
6 @app.schedule(schedule="0 0 * * *", arg_name="myTimer", run_on_startup=True,
7 use_monitor=False)
```

Logs App Insights Logs Log Level Stop Copy Clear Maximize Send us your feedback

Connected! You are now viewing logs of Function runs in the current Code + Test panel. To see all the logs for this Function, please go to 'Logs' from the Function menu.

2024-07-27T01:54:49Z [Information] Executing 'Functions.BlobCleanupFunction' (Reason:'This function was programmatically called via the host APIs.', Id=94d29444-9409-4149-a564-c4ec4b956c29)  
2024-07-27T01:54:49Z [Verbose] Sending invocation id: '94d29444-9409-4149-a564-c4ec4b956c29'  
2024-07-27T01:54:49Z [Verbose] Posting invocation id:94d29444-9409-4149-a564-c4ec4b956c29 on workerId:1aaa73eb-75dc-4671-b1dc-136f05fe98c3  
2024-07-27T01:54:49Z [Information] Python timer trigger function executed.  
2024-07-27T01:54:49Z [Information] Executed 'Functions.BlobCleanupFunction' (Succeeded, Id=94d29444-9409-4149-a564-c4ec4b956c29, Duration=16ms)

Logs App Insights Logs Log Level Stop Copy Clear Maximize Send us your feedback

Connected! You are now viewing logs of Function runs in the current Code + Test panel. To see all the logs for this Function, please go to 'Logs' from the Function menu.

2024-07-27T01:54:49Z [Information] Executing 'Functions.BlobCleanupFunction' (Reason:'This function was programmatically called via the host APIs.', Id=94d29444-9409-4149-a564-c4ec4b956c29)  
2024-07-27T01:54:49Z [Verbose] Sending invocation id: '94d29444-9409-4149-a564-c4ec4b956c29'  
2024-07-27T01:54:49Z [Verbose] Posting invocation id:94d29444-9409-4149-a564-c4ec4b956c29 on workerId:1aaa73eb-75dc-4671-b1dc-136f05fe98c3  
2024-07-27T01:54:49Z [Information] Python timer trigger function executed.  
2024-07-27T01:54:49Z [Information] Executed 'Functions.BlobCleanupFunction' (Succeeded, Id=94d29444-9409-4149-a564-c4ec4b956c29, Duration=16ms)

---

## **Testing the Project**

In this section, I will describe the final testing phase of the "ShareSafely" web application to ensure all components function as expected and the application meets the security and usability standards set during development.

### **Accessing the Deployed Application**

To begin the final testing, I accessed the "ShareSafelyApp" web application deployed on Azure Web Apps.

1. Navigating to the Application:
  - I went to the Azure Portal and navigated to the "ShareSafelyApp" Web App.
  - From the overview page, I accessed the live application at the Web App URL.

### **Testing File Upload Functionality**

The core functionality of the application, which involves uploading files and generating a time-limited access link, was thoroughly tested.

1. Uploading a File:
  - A new tab opened in the browser with the interface of the web application.
  - I clicked on the "Upload" button and selected a file named "Project Workflow Design" to upload. This file was not present in the "uploads" container before the test.
  - After clicking "Upload," the application processed the file upload.
2. Generating a Time-Limited Link:
  - Upon successful upload, the application generated a unique, time-limited link for the uploaded file.
  - A new tab opened displaying this link, which was designed to expire after 15 seconds.

### **Verification of Link Expiration**

To confirm that the security feature of the application was functioning correctly, I tested the expiration of the generated link.

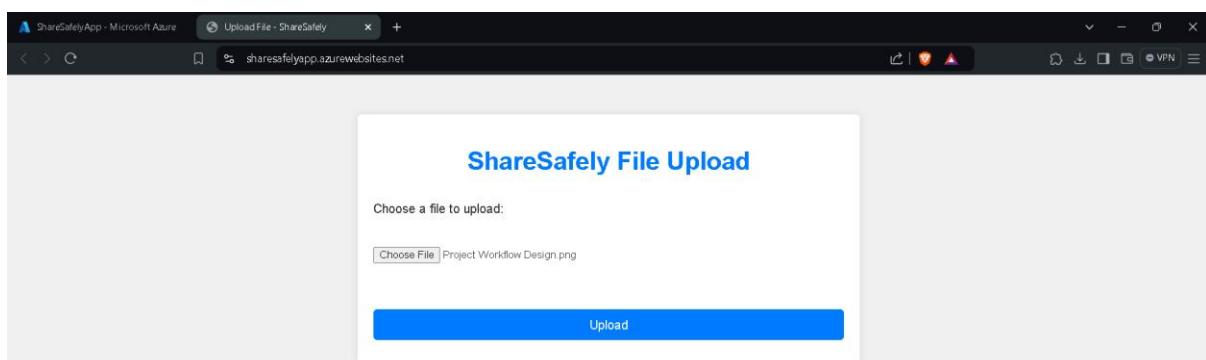
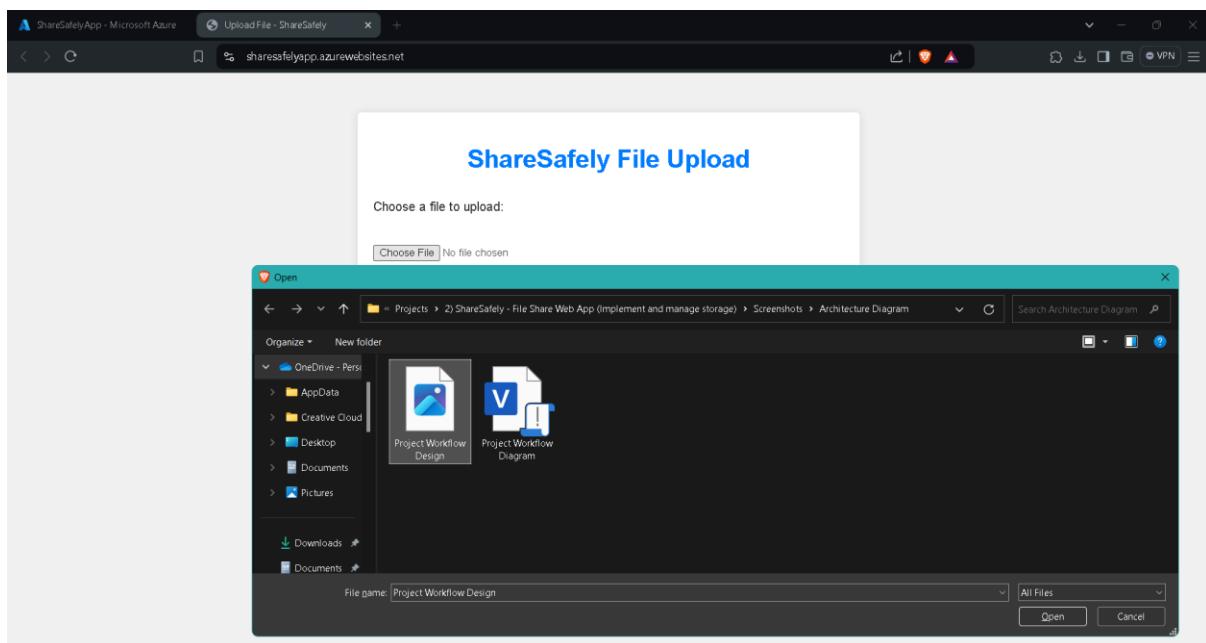
### Checking Link Accessibility:

- Immediately after the link was generated, I accessed it to verify that the file could be downloaded.
- After 15 seconds, I attempted to access the link again. As expected, the link was no longer accessible, confirming that the file could not be accessed after the expiration period.

### Screenshots:

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes 'Microsoft Azure', a search bar, and user information. Below the header, the URL is 'portal.azure.com/#@vivekvash1507@gmail.com'. The main content area displays the 'ShareSafetyApp' web app overview. On the left, there's a sidebar with navigation links like Home, Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Microsoft Defender for Cloud, Events (preview), Better Together (preview), Deployment, Performance, Settings (Environment variables, Configuration, Authentication, Application Insights, Identity, Backups, Custom domains), and JSON View. The main panel has tabs for Overview, Properties, Monitoring, Logs, Capabilities, Notifications, and Recommendations. Under the Overview tab, the 'Essentials' section lists Resource group (ShareSafetyRG), Status (Running), Location (East US), Subscription (Azure subscription 1), and Subscription ID (redacted). It also shows Default domain (sharesafelyapp.azurewebsites.net), App Service Plan (ASP-ShareSafetyRG-91b3 (F1:1)), Operating System (Linux), and Health Check (Not Configured). The 'Properties' tab shows the Web app settings: Name (ShareSafetyApp), Publishing model (Code), and Runtime Stack (Python - 3.12). The 'Deployment Center' section shows deployment logs (Successful on Friday, July 26, 08:22:45 PM, Refresh), Last deployment, and Deployment provider (None). The 'Domains' section shows the Default domain (sharesafelyapp.azurewebsites.net) and Custom domain (Add custom domain). The 'Application Insights' section indicates 'Not supported. Learn more'. The 'Networking' section is currently empty.

The screenshot shows a browser window titled 'ShareSafetyApp - Microsoft Azure' with the URL 'sharesafelyapp.azurewebsites.net'. The page title is 'Upload File - ShareSafety'. A large modal dialog box is centered, titled 'ShareSafety File Upload'. It contains the instruction 'Choose a file to upload:' and a 'Choose File' button. Below the button, it says 'No file chosen'. At the bottom of the modal is a blue 'Upload' button.



The screenshot shows the Microsoft Azure Storage Container Overview page for the 'uploads' container. The URL in the browser is [https://portal.azure.com/#view/Microsoft\\_Azure\\_Storage/ContainerMenuBlade/~/overview/storageAccountId/%2Fsubscriptions%2Fe...](https://portal.azure.com/#view/Microsoft_Azure_Storage/ContainerMenuBlade/~/overview/storageAccountId/%2Fsubscriptions%2Fe...). The page displays the following information:

- Authentication method:** Access key (Switch to Microsoft Entra user account)
- Location:** uploads
- Search:** Search resources, services, and docs (G+)
- Actions:** Upload, Change access level, Refresh, Delete, Change tier, Acquire lease, Break lease, View snapshots, Create snapshot, Give feedback.
- Overview:** Diagnose and solve problems, Access Control (IAM), Settings.
- Blob details:** Search blobs by prefix (case-sensitive), Show deleted blobs, Add filter.
- Table Headers:** Name, Modified, Access tier, Archive status, Blob type, Size, Lease state.
- No results:** No results found.

The screenshot shows the ShareSafety file upload confirmation page. The URL in the browser is [https://sharesafelyapp.azurewebsites.net/link?url=https://sharesafelystrg.blob.core.windows.net/uploads/Project+Workflow+Design.png?se=2024-07-28T03%3A47%3A52Z&sp=r&sv=2024-08-04&sr=b&sig=%2Bmyw6twN...VpHnly95C2jauOMv0nb01Xn8kik4IIP%2Bf0g%3D](https://sharesafelyapp.azurewebsites.net/link?url=https://sharesafelystrg.blob.core.windows.net/uploads/Project+Workflow+Design.png?se=2024-07-28T03%3A47%3A52Z&sp=r&sv=2024-08-04&sr=b&sig=%2Bmyw6twN...). The page displays the following message:

**File Uploaded Successfully**

Share this link to provide access to the file:

<https://sharesafelystrg.blob.core.windows.net/uploads/Project+Workflow+Design.png?se=2024-07-28T03%3A47%3A52Z&sp=r&sv=2024-08-04&sr=b&sig=%2Bmyw6twN...VpHnly95C2jauOMv0nb01Xn8kik4IIP%2Bf0g%3D>

This screenshot shows the Microsoft Azure Storage Container Overview page for the 'uploads' container. The container has one blob named 'Project Workflow Design.png'. The blob details are as follows:

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
Project Workflow Design.png	7/27/2024, 9:47:37 PM	Hot (Inferred)		Block blob	52.3 KiB	Available

This screenshot shows the Microsoft Azure Storage Container Overview page for the 'uploads' container. The blob name is now 'Project%20Workflow%20Design.png'. The page displays an XML error message indicating an authentication failure:

```
<Error>
<Script id="applocemmlbhjplcgkfciliegomcon">
<Script/>
<Script/>
<Code>AuthenticationFailed</Code>
<Message>The server failed to authenticate the request. Make sure the value of Authorization header is formed correctly including the signature. RequestId:a077c6c7-901a-004c-35a0-e8b6bd000000 Time:2024-07-28T03:47:52Z&sp=8sv...</Message>
<AuthenticationErrorDetail>Signed expiry time [Sun, 28 Jul 2024 03:48:01 GMT] must be after signed start time [Sun, 28 Jul 2024 03:47:52 GMT]</AuthenticationErrorDetail>
</Error>
```

---

## Challenges and Learnings

### Challenges Faced:

- Managing secure access and integration between Azure services.
- Resolving CORS issues during testing.

### Learnings:

- Importance of secure credential management.
  - Efficient use of Azure Monitor and Functions for automated tasks.
- 

## Conclusion

The project successfully implemented a secure and efficient file-sharing web application, showcasing skills in:

1. **Azure Blob Storage:** For storing and managing uploaded files.
2. **Azure Web Apps:** For hosting and deploying the web application.
3. **Azure Key Vault:** For securely managing sensitive credentials.
4. **Azure Functions:** For automating periodic cleanup of expired files.
5. **Azure Monitor:** For setting up metrics and alerts to monitor application performance.

And Strong **Hands-on** Capabilities in the

(Implement and manage storage)

Scope of **Microsoft Certified: Azure Administrator Associate (AZ-104)**

## Skills measured

- Manage Azure identities and governance
- **Implement and manage storage**
- Deploy and manage Azure compute resources
- Implement and manage virtual networking
- Monitor and maintain Azure resources