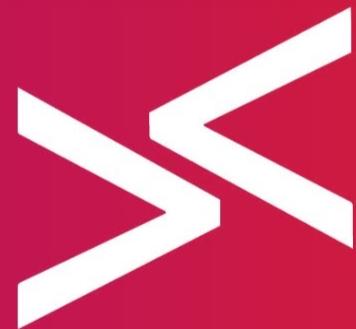


AVD Tech Fest



Live in Edinburgh

14th September 2023



AVD TechFest

Thank you to our Sponsors!

10ZiG.

control^{UP}

deviceTRUST

LIQUIT

nerdio

stratodesk

IGEL

ezeep
by ThinPrint



AI and the Art of Knowledge Base Building for Azure Virtual Desktop Support

Bridging AI and Data for a
Smoother Support Experience

Ryan Mangan

RYAN MANGAN

CEO @

Ryan is an end-user computing (EUC) specialist & Cloud technologist. An author, speaker and presenter, who has helped customers and technical communities with end-user computing solutions.



Mastering Azure Virtual Desktop

Mastering Azure Virtual Desktop

The ultimate guide to the implementation and management of Azure Virtual Desktop



Ryan Mangan

Foreword by Jim Moyle, Senior Program Manager, Azure Virtual Desktop, Microsoft





Session Disclaimer

While the Presenter has made every effort to ensure that the information in this presentation is accurate, we do not assume and hereby disclaim any liability to any party for any loss, damage, or disruption caused by errors or omissions. This holds true whether such errors or omissions result from negligence, accident, or any other cause.

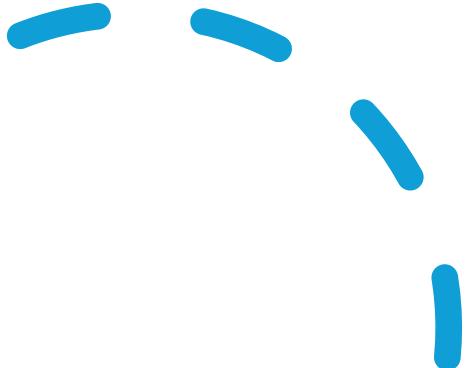


Agenda

- Introduction
 - Challenge
 - The Power of Data & AI
 - Lets Look at Using AI with AVD
 - Script Integration with OpenAI
 - Error Handling using AI
 - From Error to Knowladge
 - A Proactive Approach
 - Demonstration
 - Benefits roadmap
 - The way forward
-



Introduction

- 
- 1. The Changing Landscape of Tech Support**
 - As technology evolves, so do the expectations and demands of users.
 - Traditional support methodologies struggle to keep up with the vastness of the digital ecosystem.
 - There's an increasing need to turn to more intelligent solutions.
 - 2. Merging AI Capabilities with Human Intelligence**
 - Humans possess intuition, understanding, and the ability to empathise.
 - AI brings rapid data processing, pattern recognition, and immense knowledge.
 - Together, they create a formidable, efficient, and highly responsive support mechanism.
 - 3. Improving Support through Accumulated Knowledge**
 - Every error, every issue, every ticket is a learning experience.
 - AI allows us to accumulate this knowledge and transform it into actionable insights.
 - Future support is enhanced by drawing from these experiences, ensuring quicker resolutions and more satisfied users.

The Challenge

1. A Plethora of Potential Issues

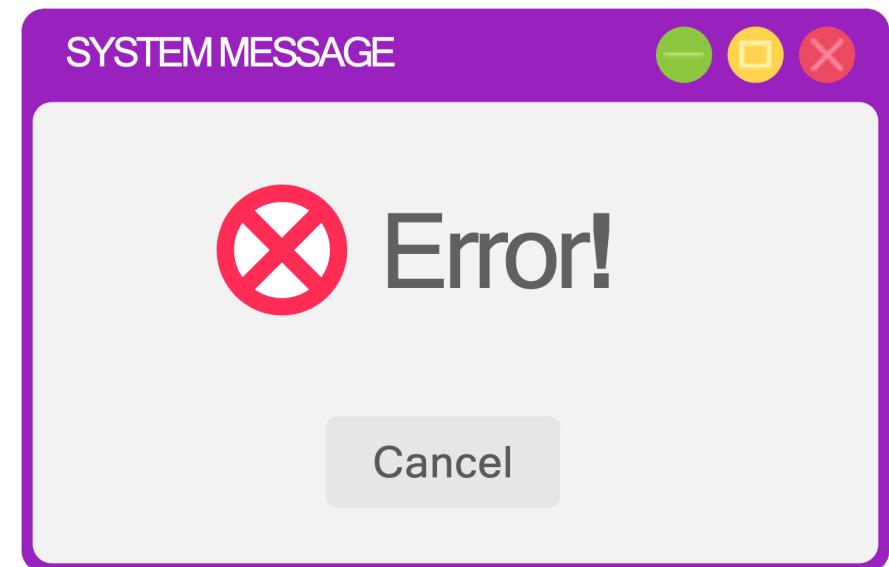
- Azure Virtual Desktop, while powerful, is susceptible to a myriad of potential glitches and errors.
- From connectivity problems to application-specific errors, the landscape of issues is vast and varied.
- The rapid adoption and diverse use cases further amplify the spectrum of potential challenges.

2. Time-Consuming Manual Troubleshooting

- Traditional troubleshooting methods often involve sifting through logs, manual error identification, and long resolution times.
- Each issue, while unique, often requires reinventing the wheel in terms of finding a solution.
- The manual process is both tedious for the support teams and frustrating for the end-users.

3. Need for a More Efficient Approach

- The current system, while effective to some extent, is far from optimal.
- As Azure Virtual Desktop continues to grow and evolve, the support mechanism must keep pace.
- There's a clear and pressing need for a smarter, faster, and more adaptive support strategy.



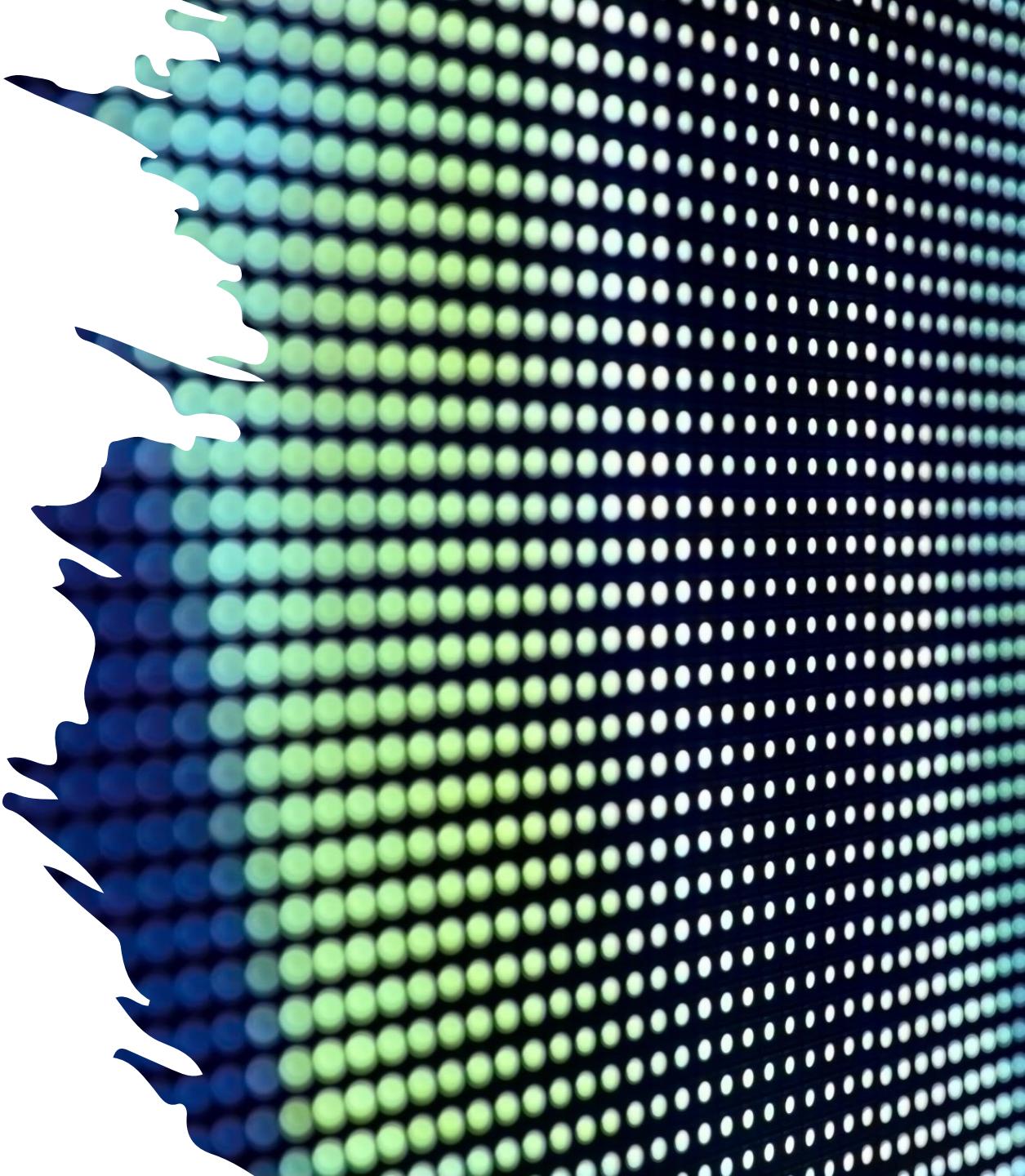


The Power of Data & AI

- 1. Transforming Every Error into a Learning Experience**
 - Each error, each glitch, is not just a roadblock but an opportunity.
 - By capturing real-time data on these errors, AI can learn, adapt, and even predict future issues.
 - Mistakes are no longer mere setbacks but valuable lessons for continuous improvement.
- 2. Predictive Analysis to Anticipate Issues**
 - Utilising AI's powerful analytical capabilities, we can foresee potential issues even before they manifest.
 - Predictive models, trained on vast amounts of historical data, help pre-empt problems, ensuring smoother user experiences.
 - By being proactive rather than reactive, we ensure a robust and resilient system.
- 3. Refining User Experience through AI-Driven Solutions**
 - AI's primary goal isn't just to solve but to enhance.
 - By offering real-time solutions, optimising system performance, and minimising downtimes, AI significantly elevates the overall user experience.
 - The future isn't about just solving problems; it's about providing a seamless, hassle-free environment for users.



Lets now look at
Using AI With AVD



RMITBLOG / AI_KB_Generator

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

AI_KB_Generator Public

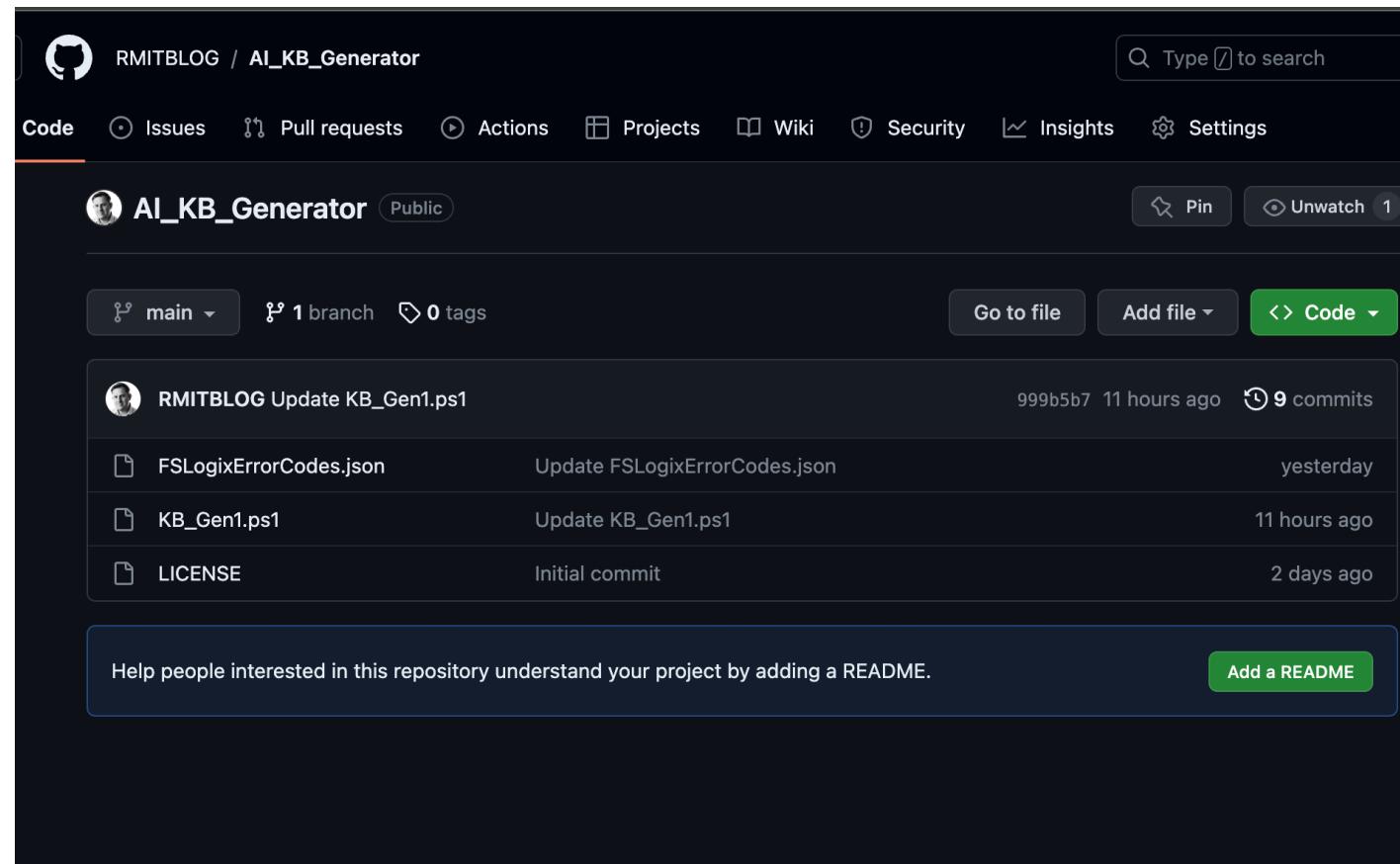
Pin Unwatch 1

main 1 branch 0 tags Go to file Add file Code

RMITBLOG Update KB_Gen1.ps1 999b5b7 11 hours ago 9 commits

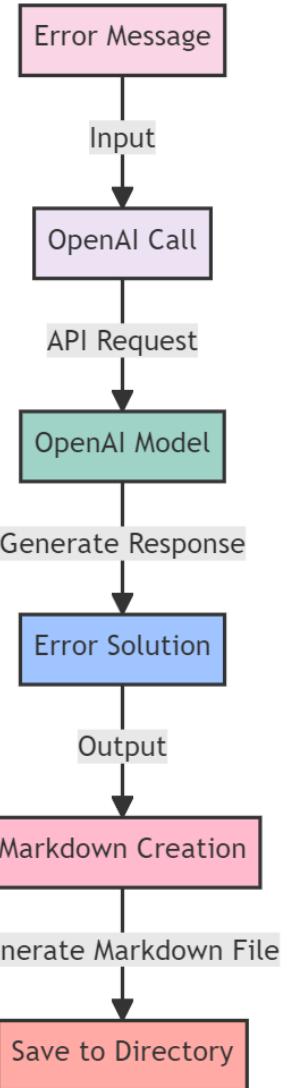
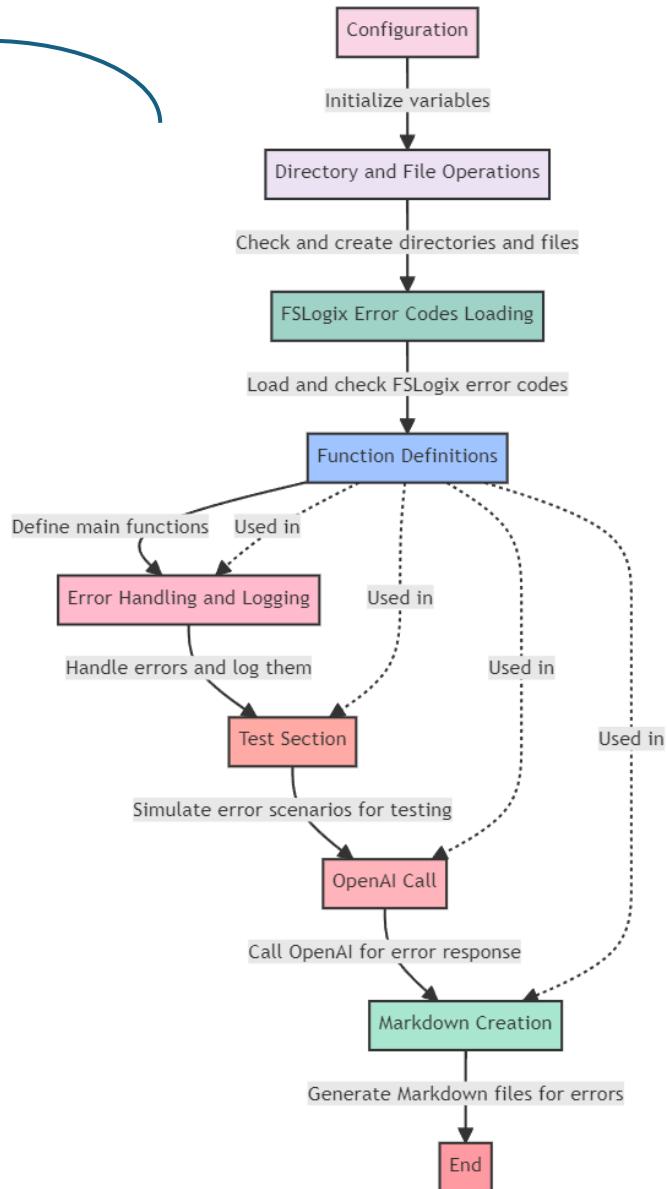
File	Commit Message	Time
FSLogixErrorCodes.json	Update FSLogixErrorCodes.json	yesterday
KB_Gen1.ps1	Update KB_Gen1.ps1	11 hours ago
LICENSE	Initial commit	2 days ago

Add a README



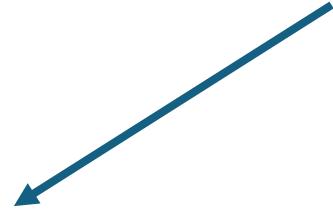
Grab the Script

Script Workflow



OpenAI Process

```
$apiEndpoint = $Endpoint
$headers = @{
    "Content-Type" = "application/json"
    "api-key"      = $APIKEY
}
$body = @{
    messages = @(
        @{
            role = "system"; content = $PreInformation }, # Add pre-information
here
        @{
            role = "system"; content = $fslogixErrorscodes }, # Add pre-information
here
            @{
                role = "system"; content = "You are an IT professional looking for
technical answers to help resolve the $Classification error." },
            @{
                role = "user"; content = "How to fix error: $Error" }
        )
    max_tokens      = 800
    temperature    = 0.7
    frequency_penalty = 0
    presence_penalty = 0
    top_p          = 0.95
    stop           = $null
} | ConvertTo-Json
```



Script Integration with OpenAI

1. Fetching Error Codes: FSLogixErrorCodes.json

- FSLogix Overview: FSLogix is a set of solutions that enhance, enable, and simplify non-persistent Windows computing environments. Within the Azure Virtual Desktop sphere, it plays a pivotal role by improving performance and centralising user profiles.
- Dynamic Error Retrieval: The script is designed to dynamically pull error codes from the FSLogixErrorCodes.json file. This ensures that the latest error definitions are always accessible, paving the way for real-time error identification and analysis.

2. Understanding Error Classifications through Patterns

Pattern Recognition: By analysing the text and context of errors, our script exhibits a keen ability to identify recurring patterns. This swift recognition is instrumental in timely and accurate error categorisation.

Mapping Errors: Once these patterns are discerned, the script efficiently maps them to specific error categories. This robust mapping process forms the bedrock upon which effective resolutions are built.

3. Engaging OpenAI for Optimal Responses

- Seamless AI Integration: With OpenAI at its core, the script transcends traditional error-handling methodologies. This integration brings about not just intelligence but also an adaptive learning approach to troubleshooting.
- Leveraging OpenAI's Acumen: Every time the script encounters an error, OpenAI dives deep into its extensive knowledge repository. Drawing from countless data points, it then recommends the most potent solution, ensuring that the troubleshooting process is not just fast but also effective.

```
"Title": "FSLogix Error Codes",
"Description": "Reason codes for use when troubleshooting FSLogix issues",
"ErrorCodes": [
  {
    "Code": 0,
    "Type": "Normal",
    "Description": "STATUS_SUCCESS",
    "Explanation": "Success"
  },
  {
    "Code": 1,
    "Type": "Error",
    "Description": "ERROR",
    "Explanation": "Can't load user's profile"
  },
  {
    "Code": 2,
    "Type": "Error",
    "Description": "ERROR_VIRT_DLL",
    "Explanation": "Virtual disk API isn't available on this machine"
  },
  {
    "Code": 3,
    "Type": "Error",
    "Description": "ERROR_GET_USER",
    "Explanation": "Can't retrieve user's security identifier"
  },
  {
    "Code": 4,
    "Type": "Error",
    "Description": "ERROR_HANDLE_ODFC",
    "Explanation": "Handle object does not exist or is not valid"
  }
]
```

```
function Handle-Error {
    param (
        [Parameter(Mandatory = $true)]
        [string]$errorMessage
    )

    $mutex.WaitOne()

    $errorsList = @()
    if (Test-Path $errorLogFile) {
        $content = Get-Content $errorLogFile | ConvertFrom-Json
        # Ensure $errorsList is always an array
        $errorsList = @($content)
    }

    # Load the existing KB register
    $kbRegister = @()
    if (Test-Path $kbRegisterFile) {
        $kbContent = Get-Content $kbRegisterFile | ConvertFrom-Json
        $kbRegister = @($kbContent)
    }

    $existingMarkdownFile = Get-ChildItem -Path $markdownDirectory -Filter "*.$(Get-FileExtension $kbRegisterFile)"
    $content = Get-Content $_.FullName
    $content -match [regex]::Escape($errorMessage)
}

if ($existingMarkdownFile) {
    Write-Host "A Markdown file for this error already exists: $($existingMarkdownFile.FullName)"
    $mutex.ReleaseMutex()
    return
}

# Check if the error message matches any FSLogix error codes
$fslogixError = $fslogixErrors | Where-Object { $errorMessage -match $_.ErrorMessage }
if ($fslogixError) {
    # Append FSLogix error information to the pre-information content
    $preInformation += "`n" + $fslogixError.Description
}
```

Error Handling Using AI

Mapping Error Patterns to Categories

- Intelligent Recognition: The prowess of AI lies in its ability to rapidly detect and categorise multifarious error patterns. By analysing code structure, contextual cues, and historical data, the AI discerns these patterns in real-time.
- Classification: A meticulous classification isn't just about labels; it's the compass guiding us to resolution. The precision in this step determines the swiftness and efficacy of the subsequent troubleshooting process.

Querying OpenAI for the Solution

- On-the-Fly Interaction: As an error rears its head, there's no manual lookup. The system instantaneously communicates with OpenAI, seeking guidance on the best remedy.
- Diving into the Knowledge Pool: OpenAI, with its vast and ever-expanding reservoir of data, scans for the most congruous solution. It's not about finding a solution, but the best solution.

Dynamic Markdown Creation for Accumulated Knowledge

- Active Documentation: Once a hurdle is overcome, the journey isn't over. The AI facilitates the creation of a markdown file detailing the error and its solution. It's like an athlete reviewing game tape; preparation for the future.
- Knowledge Vault: Over time, these markdown files burgeon into a formidable repository. This is no mere archive but a living, growing compendium ensuring that every past mistake paves the way for a faster resolution in the future.

From Error to Knowledge

Markdown Files: Rich, Formatted Knowledge Accumulation

- Living Documentation: Every error encountered isn't just a problem solved; it's knowledge gained. Each solution is meticulously documented in a markdown file, serving as an ever-growing manual.
- Continuous Evolution: These markdown files are not mere pages in an old book. They evolve, change, and grow. As new errors arise, or old ones recur in varied guises, our knowledge base adapts, ensuring that it's always relevant and up-to-date.

Proactive Support with Historical Data

- Drawing from Yesteryears: It's not about reinventing the wheel. Past errors provide insights, patterns, and early warning signs. By analysing these, we pre-empt potential problems before they become critical issues.
- Archives in Action: Our historical data isn't a dusty old library. It's an active ally, constantly accessed to provide insights, forecasts, and sometimes, a trip down memory lane to solve an enigmatic issue.

Streamlining Repetitive Issues

- The Pattern Detective: AI isn't just a solver; it's an observant detective. It identifies recurring issues, pinpoints their frequency, and traces their root causes, ensuring we're not just fighting fires but preventing them.
- Rapid Fire Solutions: Recognising recurring patterns isn't a mere academic exercise. By identifying these familiar foes, the AI enables lightning-fast resolutions. For the user, this means less downtime, fewer frustrations, and a smoother experience.



A Proactive Approach

Using Historical Data for Predictive Support

Memory Power: In the digital age, our AI's memory is its superpower. By accessing and analysing vast archives of past data, the system isn't just recalling; it's learning and improving.

Foreseeing the Future: AI's prowess lies not just in recognising past patterns but in using them to predict future glitches. By anticipating issues, we don't just react; we prepare, often stopping problems before they even start.

Minimising Downtimes and Improving User Experience

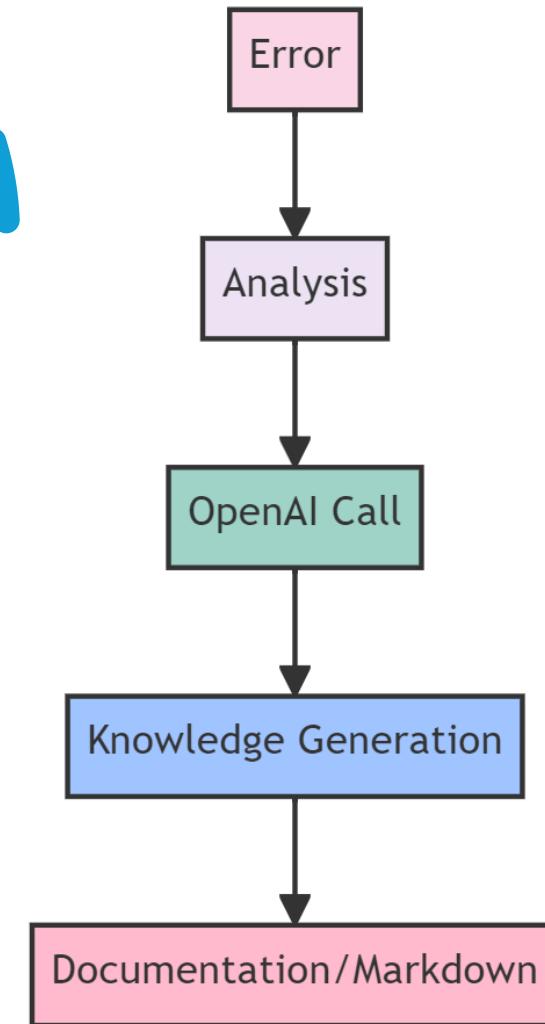
Proactive Measures: Gone are the days when we waited for things to break. With predictive insights at our fingertips, we act swiftly and decisively, heading off issues at the pass and ensuring smooth operations.

A Fluid Journey: It's not just about preventing problems; it's about enhancing the user's journey. An AI-driven experience ensures fewer interruptions, faster solutions, and a digital journey that feels smooth and intuitive.

A Shift from Reactive to Proactive Support

Redefining Support: Our tech support paradigm is evolving. Instead of merely reacting to crises, we're focusing on preventing them. It's a shift from firefighting to fire prevention, and it's revolutionising the way we approach challenges.

Looking Beyond the Horizon: While it's crucial to address the challenges of today, our vision is firmly set on tomorrow. By being proactive, we're not just solving; we're strategising, ensuring that our support mechanisms are always a step ahead.



Examining the script



Demonstration

Watch OpenAI in Action

- Real-time Querying: Transition from error analysis to solution seeking. Show how post-analysis, the script instantly reaches out to OpenAI, seeking potential fixes.
- From Analysis to Answers: The audience gets a front-row seat to AI's prowess. Display the speed and accuracy with which OpenAI understands the problem and proposes actionable solutions.

Experience the Creation of a Markdown Knowledge Snippet

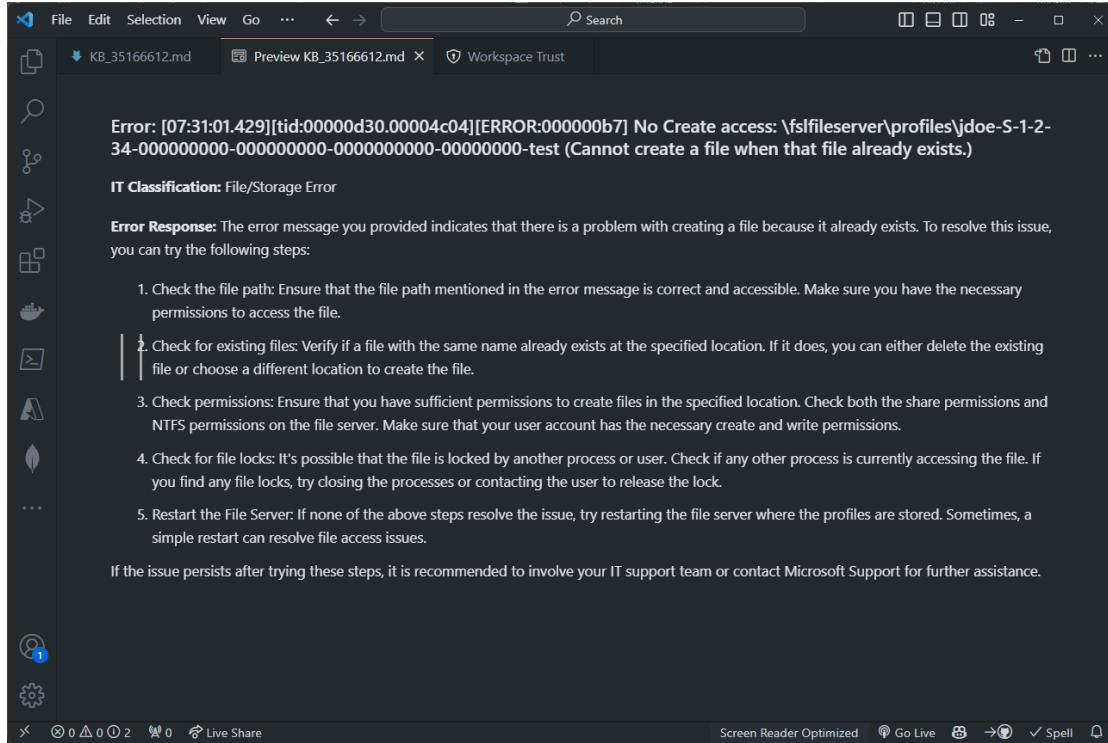
- Documentation Comes Alive: As the AI determines the solution, let the audience witness the creation of a markdown file in real-time. This showcases not just the problem-solving, but also the knowledge retention aspect of the system.
- Building Blocks of Knowledge: End on a forward-thinking note. Stress upon how every such markdown file isn't just a one-off solution but a brick in the ever-growing edifice of our knowledge repository.

Demo

The screenshot shows a Windows desktop environment with two open windows:

- Administrator: Windows PowerShell ISE**: A window titled "KBGen.ps1" containing PowerShell script code. The code is used to generate error messages for testing. It includes sections for handling errors and generating sample error messages. The output pane shows several generated error messages, such as "[11:35:59.241][tid:00000d30.00004f38][ERROR:80070003] Failed to save installed AppxPackages (The system can't find the file specified)".
- KBCollection**: A File Explorer window showing a folder structure. The path is "This PC > Local Disk (C:) > temp > KBCollection". The folder is empty, as indicated by the message "This folder is empty."

Markdown Files



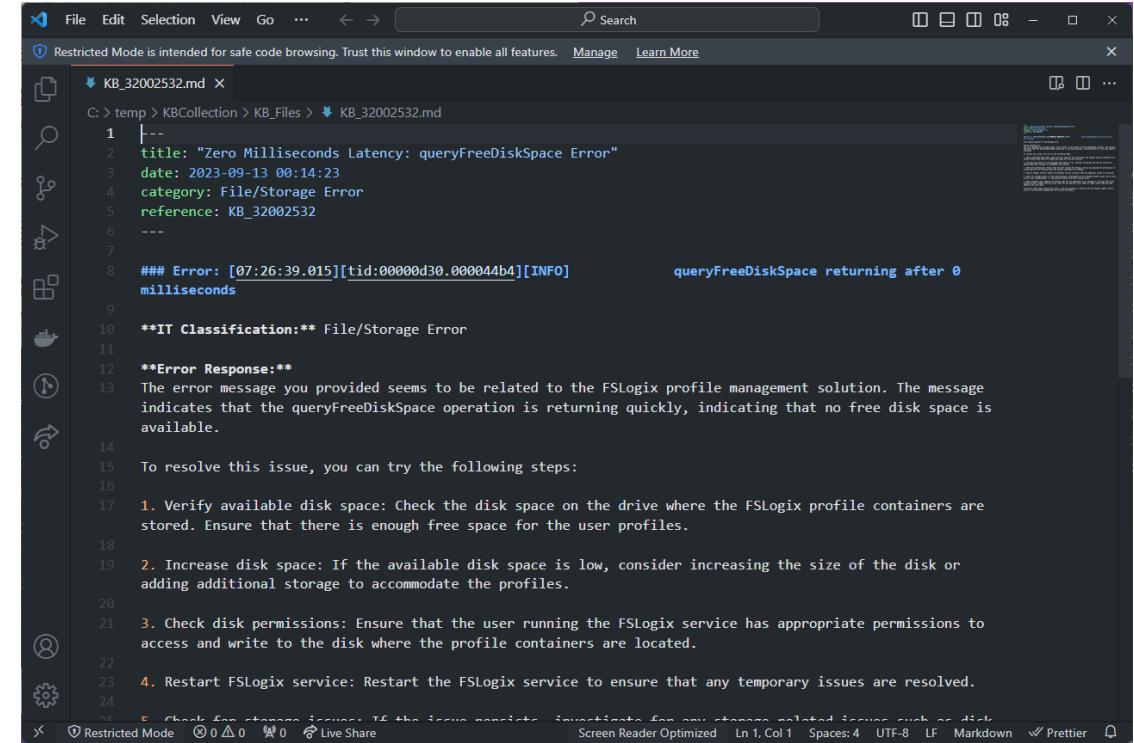
Error: [07:31:01.429][tid:00000d30.00004c04][ERROR:00000b7] No Create access: \fsfileserver\profiles\jdoe-S-1-2-34-00000000-00000000-00000000-0000000-test (Cannot create a file when that file already exists.)

IT Classification: File/Storage Error

Error Response: The error message you provided indicates that there is a problem with creating a file because it already exists. To resolve this issue, you can try the following steps:

1. Check the file path: Ensure that the file path mentioned in the error message is correct and accessible. Make sure you have the necessary permissions to access the file.
2. Check for existing files: Verify if a file with the same name already exists at the specified location. If it does, you can either delete the existing file or choose a different location to create the file.
3. Check permissions: Ensure that you have sufficient permissions to create files in the specified location. Check both the share permissions and NTFS permissions on the file server. Make sure that your user account has the necessary create and write permissions.
4. Check for file locks: It's possible that the file is locked by another process or user. Check if any other process is currently accessing the file. If you find any file locks, try closing the processes or contacting the user to release the lock.
5. Restart the File Server: If none of the above steps resolve the issue, try restarting the file server where the profiles are stored. Sometimes, a simple restart can resolve file access issues.

If the issue persists after trying these steps, it is recommended to involve your IT support team or contact Microsoft Support for further assistance.



```
C: > temp > KBCollection > KB_Files > KB_32002532.md
```

1 |--
2 title: "Zero Milliseconds Latency: queryFreeDiskSpace Error"
3 date: 2023-09-13 00:14:23
4 category: File/Storage Error
5 reference: KB_32002532
6 ---
7
8 *** Error: [07:26:39.015][tid:00000d30.00004b4][INFO] queryFreeDiskSpace returning after 0 milliseconds
9
10 **IT Classification:** File/Storage Error
11
12 **Error Response:**
13 The error message you provided seems to be related to the FSLogix profile management solution. The message indicates that the queryFreeDiskSpace operation is returning quickly, indicating that no free disk space is available.
14
15 To resolve this issue, you can try the following steps:
16
17 1. Verify available disk space: Check the disk space on the drive where the FSLogix profile containers are stored. Ensure that there is enough free space for the user profiles.
18
19 2. Increase disk space: If the available disk space is low, consider increasing the size of the disk or adding additional storage to accommodate the profiles.
20
21 3. Check disk permissions: Ensure that the user running the FSLogix service has appropriate permissions to access and write to the disk where the profile containers are located.
22
23 4. Restart FSLogix service: Restart the FSLogix service to ensure that any temporary issues are resolved.
24

The screenshot shows a code editor window with a dark theme. The title bar includes standard icons for File, Edit, Selection, View, Go, and a search bar. A message at the top states: "Restricted Mode is intended for safe code browsing. Trust this window to enable all features." with links to "Manage" and "Learn More". The left sidebar contains various icons for file operations like Open, Save, Find, and Refresh. The main pane displays a JSON file named KBRegister.json. The file content lists several error entries, each containing a title, category, and KB number. The errors include Process Not Found, Error: File Not Found, Service Not Found, Error 0x87D00207, Access Error: Cannot Create File - File Already Exists, and LoadProfile Failed - FrxStatus 31: Cannot Create a File When That File Already Exists.

```
1  {
2    "title": "Process Not Found: \\\"NonExistentProcess1\\\"", "category": "Process/Software Error",
3    "KB": "KB_75381407"
4  },
5  {
6    "title": "Error: File Not Found", "category": "File/Storage Error",
7    "KB": "KB_86445915"
8  },
9  {
10   "title": "Service Not Found: \u0027NonExistentService1\u0027", "category": "Service/Network Error",
11   "KB": "KB_63989458"
12 },
13 {
14   "title": "\\\"Error 0x87D00207: Unable to Install or Update Software\\\"", "category": "Process/Software Error",
15   "KB": "KB_10023818"
16 },
17 {
18   "title": "Access Error: Cannot Create File - File Already Exists", "category": "File/Storage Error",
19   "KB": "KB_35166612"
20 },
21 {
22   "title": "\\\"LoadProfile Failed - FrxStatus 31: Cannot Create a File When That File Already Exists\\\"", "category": "FsLogix Error",
23   "KB": "KB_80532392"
24 }
```

```
1  [
2    {
3      "timestamp": "2023-09-13 00:11:49",
4      "error": "Cannot find a process with the name \\\"NonExistentProcess1\\\". Verify the process name and ca",
5      "errorCategory": "Process/Software Error",
6      "errorResponse": "This error typically occurs when you are trying to perform an action on a process t",
7      "KB": "KB_75381407"
8    },
9    {
10      "timestamp": "2023-09-13 00:12:00",
11      "error": "Cannot find path \\u0027C:\\\\NonExistentFile1.txt\\u0027 because it does not exist.",
12      "errorCategory": "File/Storage Error",
13      "errorResponse": "The error message you are receiving indicates that the specified file path does not",
14      "KB": "KB_86445915"
15    },
16    {
17      "timestamp": "2023-09-13 00:12:14",
18      "error": "Cannot find any service with service name \\u0027NonExistentService1\\u0027.",
19      "errorCategory": "Service/Network Error",
20      "errorResponse": "To fix the error \\\"Cannot find any service with service name \\u0027NonExistentServi",
21      "KB": "KB_63989458"
22    },
23    {
24      "timestamp": "2023-09-13 00:12:33",
25      "error": "Error encountered with code 0x87D00207",
26      "errorCategory": "Process/Software Error",
27      "errorResponse": "Error code 0x87D00207 typically occurs when there is an issue with the installation",
28      "KB": "KB_10023818"
29    },
30    {
31      "timestamp": "2023-09-13 00:12:47"
```

KB Register

Log File



Benefits Roundup

Faster Issue Resolution

Traditional vs. AI: Juxtapose the time taken for traditional support to resolve an issue against the lightning-fast efficiency of AI. Emphasize that with AI, gone are the days of lengthy downtimes and drawn-out troubleshooting.

Instant Analytics: Highlight that AI isn't just about quick fixes but instantaneous and profound analysis, which makes the fixes more precise and relevant.

Rich, Centralised Knowledge Base

Ever-evolving Documentation: Speak about how every error encountered and every solution implemented contributes to the knowledge base. Unlike static databases, this one learns and grows.

Accessible Intelligence: While it's centralised, it's also universally accessible. Whether it's a newbie or a veteran in the support team, everyone has the same rich data at their fingertips.

Empowered Support Teams

Beyond Troubleshooting: Point out that with AI handling the heavy lifting, the support team is freed up to focus on more critical, strategic tasks. Their roles evolve beyond just problem-solving.

Confidence Boost: Having AI as an ally means the support teams operate with increased confidence. They know that they have a robust backup, ready to jump in with solutions when things get complex.

The Way Forward

Continuous Learning & Adaptation

- Ever-evolving Intelligence: Highlight that the integration of AI is not a one-time setup but an ongoing journey. As more data pours in, the AI continually refines its understanding, becoming more adept at its role.
- Feedback Loop: Discuss the importance of feedback in this journey. As solutions are implemented and results are seen, the AI learns from this feedback, ensuring better outcomes with each iteration.

Expanding AI's Role in Troubleshooting

- Beyond Error Detection: Delve into how the future might see AI playing a role not just in identifying and resolving errors but possibly in predicting and preventing them before they even occur.
- Holistic Integration: Talk about potential plans to integrate AI more deeply into the entire Azure Virtual Desktop environment, making AI's insights and capabilities available at every stage of the user experience.

Building a Resilient and Robust Support Ecosystem

- Unified Support System: Envision a future where all elements of the support system, from AI to human teams to the end-users, are interconnected, sharing insights, feedback, and solutions in real-time.
- Bounce-back Capability: Discuss the ambition to build a system so robust that even in the face of unexpected disruptions, the recovery is swift and seamless, ensuring minimal impact on the end-users.



Sustainable AI for IT Professionals

Sustainable Scripting with AI:

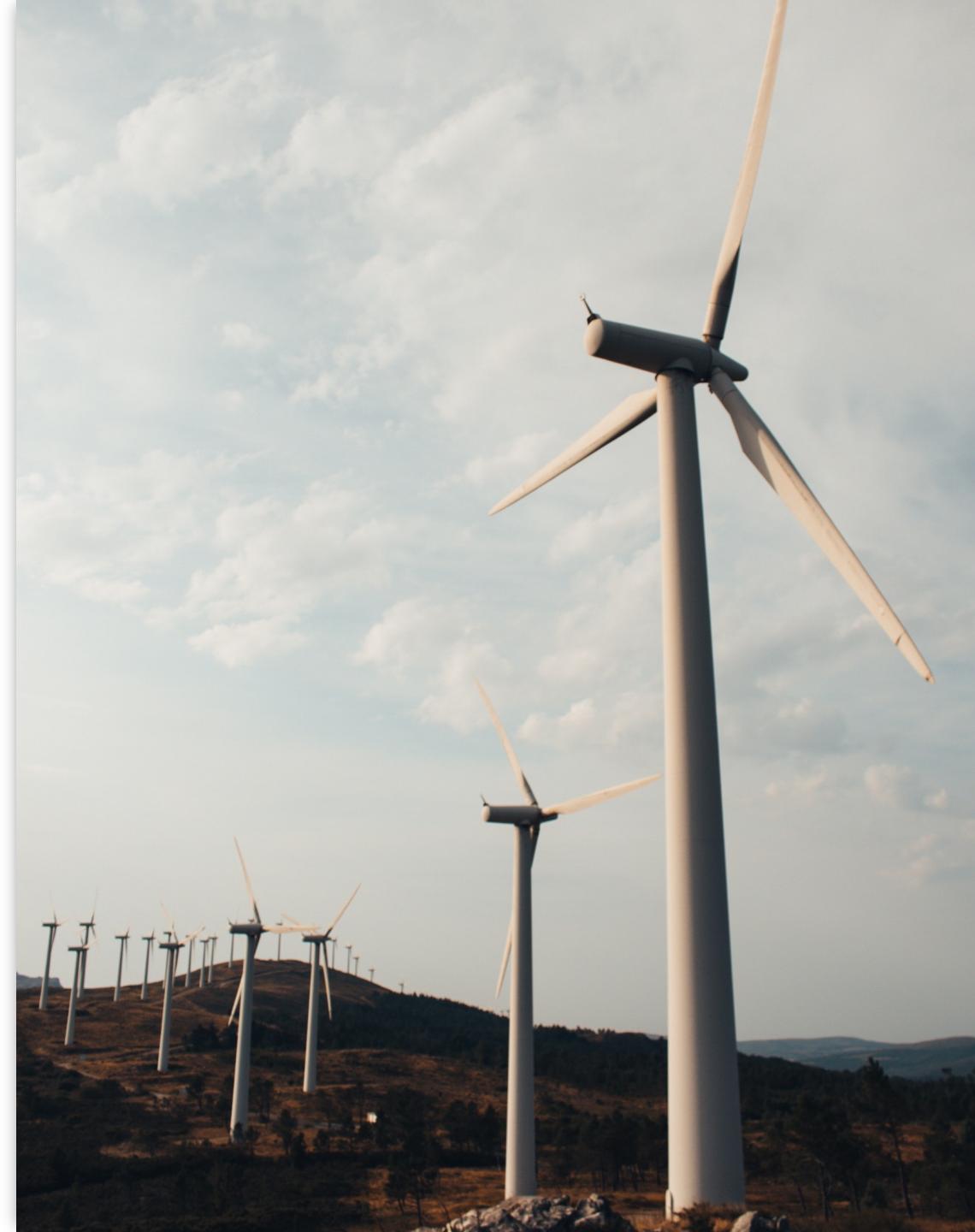
- Efficient Code Execution: Harness AI to optimise scripts, reducing computational loads and environmental footprints.
- Maximise Reusability: Design scripts modularly to avoid repeated AI interactions, conserving resources.

Storing & Referencing AI Outputs:

- Cache Intelligence: After leveraging OpenAI, store results for easy reference, eliminating the need for repeated AI invocations.
- Cost & Energy Saving: Save both finances and power by minimising unnecessary AI requests.

Proactive AI Integration:

- Anticipate Frequent Tasks: With AI's help, detect common tasks and automate, reducing constant AI interactions.
- Smart Data Management for Scripts: Use AI's insights to efficiently manage script versions, ensuring streamlined systems.



Questions



AVD TechFest

THANK YOU

Thank you to our Sponsors!

10ZiG.

controlUP

deviceTRUST

IGEL

LIQUIT

nerdio

stratodesk

ezeep
by ThinPrint