

WIPRO NGA Program – DWS

Capstone Project Presentation – 17th June & 18th
June 2025

Project Title Here – Insert Custom Actions in MSI

Presented by – Khushi Pandey

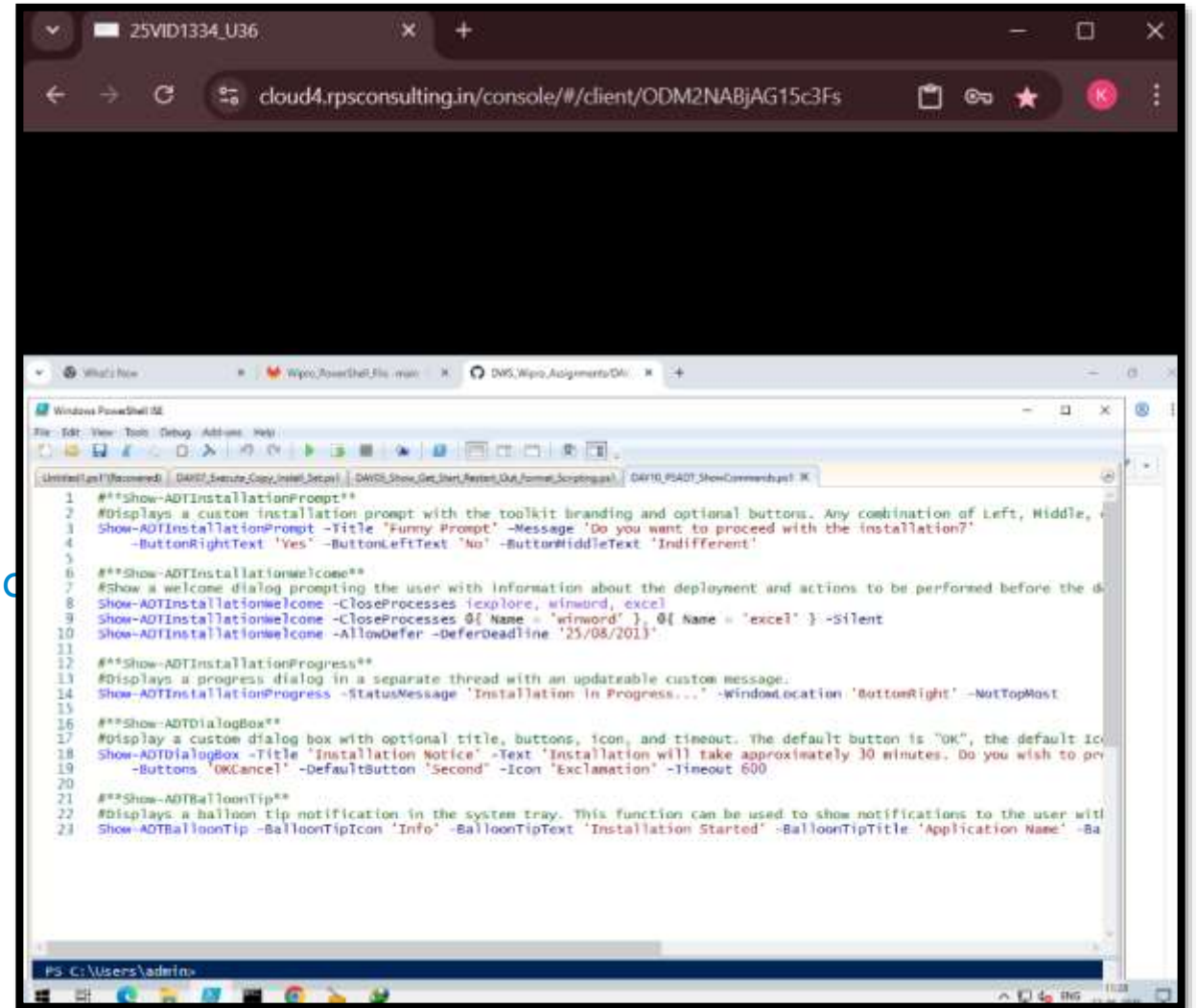
INTRODUCTION

- Custom actions can be added to an MSI installer to run PowerShell scripts, CMD commands, executables, or other logic either before, during, or after the installation or uninstallation process.
- This approach offers powerful flexibility for customizing application deployment, but it should be used cautiously to avoid potential system issues.



Project Requirement

- Windows Operating System(Windows 10/11)
- PowerShell ISE or PowerShell Console
- PowerShell App Deployment Toolkit (PSADT)
- Basic Knowledge of scripting in PowerShell
- Optional : Notepad++ or Visual Studio Code for script editing

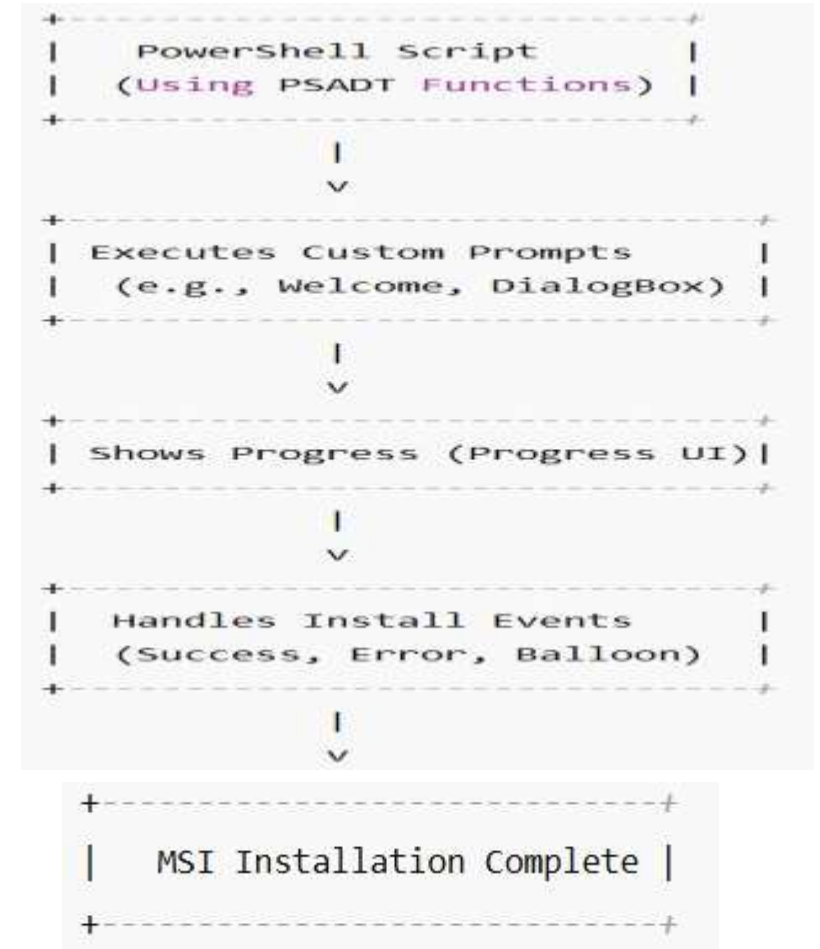


Tools and Technologies Used

- PowerShell & ISE – An essential Integrated Development Environment providing a robust environment for writing PowerShell scripts efficiently
- PSAppDeployToolkit – A comprehensive toolkit that significantly enhances PowerShell's capabilities by providing pre-built functions.
- Windows 10 VM - Safe testing
- Task Manager/System Tray – Utilize these system components to effectively test and confirm the proper display and behavior of visual notifications, including critical balloon tips

Architecture Diagram

MSI → PowerShell Script → PSADT → Execution → GUI +
Deployment Actions



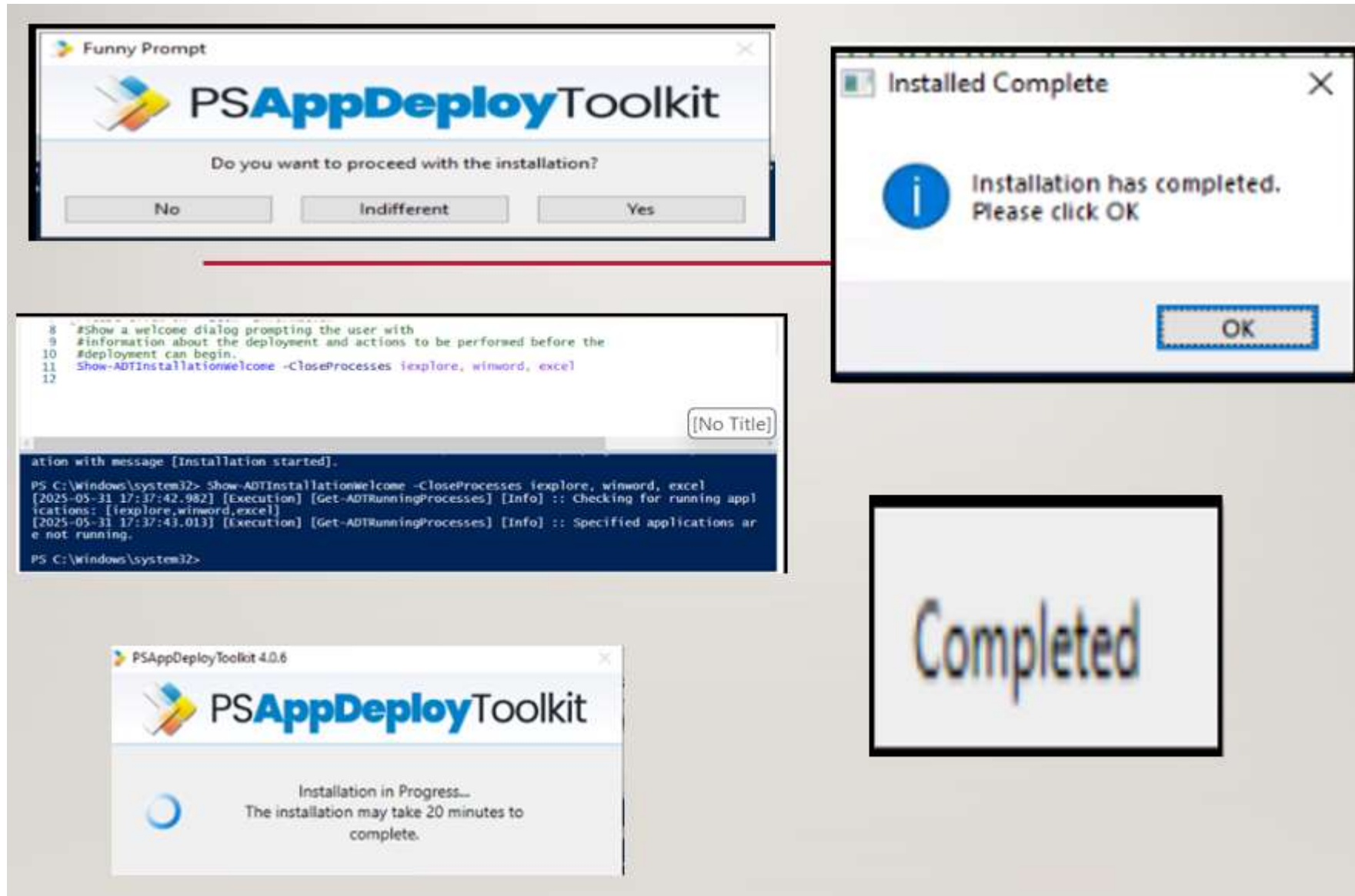
Usage Process

- Install PSADT and extract it to a working directory
- Open PowerShell ISE and load the Add-AppDevPackage.ps1 script
- Modify the script to insert custom GUI interactions using these PSADT commands
- Run the script in PowerShell to test custom actions
- Verify behaviour during install, uninstall or

Execution Overview

- Show-ADTInstallationPrompt - Displays a **custom interactive prompt** window (GUI) with a message and up to three buttons (e.g., Yes/No/Cancel or custom text).
- Show-ADTInstallationWelcome - Displays a **welcome dialog box** with instructions or warnings and automatically detects if specific applications need to be closed before continuing.
- Show-ADTInstallationProgress - Displays a **progress window** with a customizable message showing that installation is in progress.
- Show-ADTDialogBox- Displays a **custom dialog box** with OK/Cancel buttons, custom icons, and optional timeouts.
- Show-ADTBalloonTip - Displays a **balloon notification** in the Windows system tray with a message, icon, and title.

Output



Task Performed

- Created scripts for user prompts and feedback
- Example: Show-ADTInstallationPrompt with message and buttons
- Close conflicting apps before install using Show-InstallationWelcome

Function Breakdown

- Show-ADTInstallationPrompt: User choice
for setup
- Show-ADTInstallationWelcome: Closes apps
safely
- Show-ADTInstallationProgress: Status bar
- Show-ADTDialogBox: Important warnings
- Show-ADTBalloonTip: Non-intrusive
notifications

Analysis Result

Execution: Prompts and logic worked as expected

User Interaction: Smooth experience

Performance: Low resource use, stable deployment

Resources/References

- <https://github.com/PSAppDeployToolkit/PSAppDeployToolkit>
- <https://psappdeploytoolkit.com/docs/reference/functions/Show-ADTInstallationPrompt>
- <https://openai.com/index/chatgpt/>
- <https://learn.microsoft.com/>