### Assessment for **Windows Autopilot Automation Engineer**

#### Section 1: Technical Skills (60 points)

1. **PowerShell Proficiency** (15 points)

**- Provide a script that automates the deployment of a set of applications on a Windows 10 machine.**

```
$VerbosePreference = "Continue"

$apps = @("App1", "App2", "App3")

foreach ($app in $apps) {
    Try {
        Write-Host "Installing $app..."
        Start-Process "msiexec.exe" -ArgumentList "/i $app.msi /quiet /norestart" -
Wait -ErrorAction Stop -Verbose
        Write-Host "$app has been installed successfully."
    }
    Catch {
        Write-Host "Error installing $app: $($_.Exception.Message)"
    }
}
```

**- How would you debug and troubleshoot a PowerShell script that is failing intermittently?**

- **Use the -Verbose parameter** to obtain more detailed information during script execution.
- **Add exception handling with Try/Catch** blocks to capture errors.
- **Review system logs (Event Viewer)** to gather clues about the failure.
- **Run the script in different environments** and check if the issue is related to specific conditions (memory, network, permissions).

2. **PowerShell and Python Scripting** (10 points)

- **Given a scenario where you need to integrate Python for specific Automation tasks in a PowerShell-heavy environment, outline how you would approach this task.**

- **Use Python for tasks** where its performance or capabilities surpass PowerShell.
- **Call Python scripts from PowerShell** using Start-Process or Invoke-Expression.
- **Exchange data between scripts through temporary files**, standard output, or REST APIs.

- **Write a Python script that checks the system memory and reports any usage over a threshold of 75%.**

```python
import psutil

memory = psutil.virtual_memory()
if memory.percent > 75:
    print(f"Warning: Memory usage is at {memory.percent}%")
else:
    print(f"Memory usage is within the threshold ({memory.percent}%)")
```

3. **Microsoft Intune, SCCM, and Autopilot Experience** (10 points)

- **Explain how you would configure and manage devices using Microsoft Autopilot.**

- **Register** devices in Azure AD.
- **Create an Autopilot profile** in Microsoft Endpoint Manager.
- **Assign devices to the profile** for automatic configuration at startup.

- **How do Intune and SCCM differ when it comes to device management? In what situations would you prefer one over the other?**

- **Intune**: Cloud-based management, ideal for mobile devices and remote environments.
- **SCCM**: On-premises solution, suited for large organizations with advanced deployment needs.
- **Preference**: Use Intune for remote devices and SCCM for large local networks.

4. **Microsoft Active Directory and Group Policy** (10 points)

**- How would you implement Group Policy to manage devices in an organization?**

- **Create a new GPO** in the Group Policy Management Console.
- **Configure the desired policies** (security restrictions, software settings).
- **Link the GPO to an Organizational Unit** (OU) in Active Directory.

**- Provide an example where you had to troubleshoot a Group Policy Object (GPO) that wasn't applying correctly.**

- **Use the gpresult /r command** to view applied GPOs.
- **Check if the GPO** is correctly linked to the OU.
- **Verify GPO permissions** and check for conflicts with other GPOs.

5. **Application and Automation Experience** (10 points)

**- Describe your experience with integrating security and identity tools (OKTA Verify) into device setup processes.**

- **Integrating OKTA Verify:** Incorporate OKTA Verify into device setup with scripts that handle multi-factor authentication (MFA).

**- Explain how you would streamline the device setup experience to improve user productivity.**

- **Automate initial configurations** (software installation, network setup).
- **Create custom scripts** to minimize user interaction.
- **Implement predefined configuration profiles** based on user roles.

**6. **Familiarity with PowerShell Studio** (5 points)**

**- How would you use PowerShell Studio to enhance your PowerShell scripts for UI development?**

- **Enhancing Scripts with PowerShell Studio:** Create a graphical interface for complex scripts to improve user interaction.

**- Provide an example where PowerShell Studio improved your workflow.**

- **Using PowerShell Studio to develop a GUI** that allows users to select applications for installation via checkboxes.

#### **Section 2: Role-Specific Scenarios (30 points)**

**1. **Device Setup Process Enhancement** (15 points)**

**- You are tasked with enhancing the current device setup process that relies heavily on PowerShell. Describe how you would approach this.**

- **Analyze the current process** to identify inefficiencies.
- **Break the process** into reusable modules.
- **Create a simple UI for users** to select configuration options.

**- Provide a step-by-step explanation of how you would implement front-end UI development for this purpose, making use of PowerShell Studio if needed.**

- **Use PowerShell Studio to create a form with buttons**, checkboxes, and text fields.
- **Link UI actions to existing PowerShell scripts** to perform tasks such as software installation or network configuration.

**2. **Microsoft Autopilot Decoupling from Intune** (10 points)**

**- What challenges might arise when Microsoft Autopilot is decoupled from Intune? How would you address these?**

- **Loss of remote management capabilities.**
- **Increased complexity in managing devices locally without Intune.**

**- Suggest improvements to the device provisioning process using Autopilot in a scenario where Intune is not part of the workflow.**

- **Use PowerShell scripts and other local management tools** like SCCM or automated scripts to fill the gap left by Intune.
- **Improve device provisioning through automation** of local configuration tasks (software installation, policy setup).

**3. **Seamless Device Setup Experience** (5 points)**

**- Identify areas where the current setup process might cause friction for users, and describe how you would create a more seamless device provisioning experience.**

- **Automate as many configurations as possible** (profiles, security policies).
- **Reduce manual interaction during provisioning** (credential input, configuration choices).
- **Implement solutions such as Single Sign-On** (SSO) for automatic authentication.

#### **Section 3: Soft Skills and Problem Solving (10 points)**

**1. **Collaboration and Communication** (5 points)**

**- How do you typically collaborate with other engineers and stakeholders on large automation projects?**

- **Use tools like Microsoft Teams, Slack, and Git** to manage collaborative projects, maintaining open and transparent communication with the team.

**- Provide an example of how you communicated a complex technical solution to a non-technical audience.**

- **During a process automation project**.
  I explained the benefits of the scripts to a non-technical team using charts to show improvements in efficiency and error reduction.

**2. **Troubleshooting and Problem-Solving** (5 points)**

**- Describe a time when you encountered a complex technical issue related to device provisioning or automation. How did you resolve it?**

- **A device failed to provision correctly due to a security policy error**.
  I resolved it by analyzing Intune logs and correcting an issue in the device profile configuration.

**- When automating processes, how do you ensure that the automation Remains maintainable and scalable?**

- **I ensure that automation scripts are modular and reusable**, documenting each section thoroughly, and structuring the scripts so that other engineers can easily modify them.