# Master Application Packaging Validation Checklist (Enterprise Grade)

## Tab 1: 📝 Pre-Packaging & Discovery (Input/Design)

This tab captures all necessary design decisions, licensing, and application characteristics.

| ID | Item Description | Verification Method | Status (Y/N) | Details/Notes | Packager/Tester |
| --- | --- | --- | --- | --- | --- |
| **D.1.01** | **Application Type:** Identify if the application is Standard Desktop, System Service, Browser Extension, or VDI/Terminal Server specific. | Vendor Docs |  |  |  |
| **D.1.02** | **Install Behavior:** Confirm if the installation is **Per-Machine** (System) or **Per-User**. *Per-Machine is mandatory for enterprise.* | Trial Run/Reg Check |  |  |  |
| **D.1.03** | **License Info:** Document the licensing model (Volume, KMS, Serial Key, License File, Floating). | Vendor Docs |  |  |  |
| **D.1.04** | **License Application:** Document the exact method used to apply the license silently during install (e.g., MST, config file copy, registry write). | Packaging Tool |  |  |  |
| **D.1.05** | **Pre-Requisites:** Identify all external dependencies (.NET, VC++ Redistributables, Java Runtime, etc.) and their specific versions. | Vendor Docs |  |  |  |
| **D.1.06** | **Unwanted Components:** List all components that must be suppressed or removed (e.g., auto-updaters, third-party toolbars, telemetry services). | Manual Testing |  |  |  |
| **D.1.07** | **Vendor-Required Restart:** Confirm if a restart is required by the vendor's base installer, or if it can be suppressed entirely. | Log Check |  |  |  |
| **D.1.08** | **Application Architecture:** Confirm 64-bit or 32-bit (to correctly check Program Files vs. Program Files (x86)). | File Properties |  |  |  |

## Tab 2: 📦 Package Build & PSADT Configuration

This tab validates the technical integrity of the package build, specifically focusing on the **PowerShell App Deployment Toolkit (PSADT)** wrapper.

| ID | Item Description | Status (Y/N) | PSADT Command Used (Install) | PSADT Command Used (Uninstall) |
| --- | --- | --- | --- | --- |
| **P.2.01** | **PSADT Pre-Installation:** All logic verified: app processes closed (Show-InstallationWelcome -CloseApps), elevated rights confirmed. |  |  |  |
| **P.2.02** | **PSADT Installation Logic:** The primary installer execution is correctly wrapped (e.g., Execute-MSI, Execute-Process), using full silent switches. |  |  |  |
| **P.2.03** | **PSADT Post-Installation:** All customization logic verified (e.g., license copy, settings import, update service disablement). |  |  |  |
| **P.2.04** | **Exit Code Handling:** The primary command is wrapped with Execute-Process or Execute-MSI to catch **ALL** vendor-specific success codes (e.g., 3010, 123). |  |  |  |
| **P.2.05** | **PSADT Uninstall Logic:** The uninstall command is correctly defined and includes cleanup of custom registry entries or folders. |  |  |  |
| **P.2.06** | **PSADT Cleanup:** Custom cleanup code for post-uninstall remnants (e.g., ProgramData, desktop files) is included and verified. |  |  |  |
| **P.2.07** | **Versioning Metadata:** The variables $appVendor, $appName, $appVersion, and $appScriptVersion are accurately updated in the script header. |  |  |  |
| **P.2.08** | **Code Signing:** The Deploy-Application.ps1 script is digitally signed (if corporate security policy mandates it). |  |  |  |

## Tab 3: 🛠️ Exit Code Handling & Findings

This tab acts as a central repository for documenting all valid exit codes and tracking packaging issues found during testing.

| ID | Origin (Vendor/System) | Code | Description | Handling Action in PSADT | Finding Status (New/Resolved) |
| --- | --- | --- | --- | --- | --- |
| **E.3.01** | Windows Installer | 0 | Success | **Default Success** | N/A |
| **E.3.02** | Windows Installer | 3010 | Success/Reboot Required | **Added to Exit-Process list** (Reboot Suppressed) | N/A |
| **E.3.03** | Vendor EXE | 123 | Application Already Installed | **Added to Exit-Process list** (123, 0) | N/A |
| **F.3.01** | Test Run 1 | Shortcut Remains | Remove-Shortcuts missing a critical path. | **Action:** Added C:\Users\Public\Desktop to the command arguments. | **Resolved** |
| **F.3.02** | Test Run 3 | Service Fails to Start | Service configured for *Disabled* but PSADT did not modify the start type. | **Action:** Added Set-Service -Name "AppService" -StartupType Automatic -Status Running to Post-Install. | **Resolved** |
| **F.3.03** | UAT (Citrix) | User Profile Bloat | Application wrote large data to Roaming Profile. | **Action:** Added Registry entry to redirect AppData to %LocalAppData%. | **New/Pending** |

## Tab 4: 🔑 ARP & System Installation

This tab focuses on the silent installation as the **System Account** and validates the resulting system entries, which are critical for inventory and management.

| ID | Test Scenario | Execution Context | Status (Pass/Fail) | Actual Exit Code | Notes/Errors |
| --- | --- | --- | --- | --- | --- |
| **T.4.01** | **System Silent Install:** Execute install command via **PSExec -s -i** on a clean VM. | VM (System) |  |  |  |
| **T.4.02** | **Directory Check:** Verify installation path (e.g., %ProgramFiles%\App Name) exists and contains all expected files. | File System |  |  |  |
| **T.4.03** | **ARP Key Exists:** Verify the main Uninstall key exists (HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{GUID or AppName}). | Reg Check |  |  |  |
| **T.4.04** | **ARP DisplayName/Version:** Verify DisplayName, DisplayVersion, and Publisher are accurate in the ARP entry. | Reg Check |  |  |  |
| **T.4.05** | **ARP UninstallString:** Verify UninstallString contains the correct, silent command (e.g., Deploy-Application.exe Uninstall). | Reg Check |  |  |  |
| **T.4.06** | **Service/Driver Check:** Verify any background services or drivers are installed with the correct startup type (e.g., Automatic, Manual). | Services.msc/Reg |  |  |  |
| **T.4.07** | **UAC Manifest:** Confirm the main executable does **not** contain an elevation manifest (requireAdministrator). | File Properties |  |  |  |

## Tab 5: 👩‍💻 User & Shortcut Validation

This tab confirms the application is fully functional and respects the environment for a **Standard User**.

| ID | Test Scenario | User Account Type | Status (Pass/Fail) | Observation/Details |
| --- | --- | --- | --- | --- |
| **A.5.01** | **Application Launch:** Standard User launches the application successfully without UAC or admin credentials. | Standard User |  |  |
| **A.5.02** | **Core Functionality:** Standard User verifies core features (Open, Save, Network functions) work correctly. | Standard User |  |  |
| **A.5.03** | **License Activation:** Verify the application is licensed correctly upon first launch for the standard user. | Standard User |  |  |
| **A.5.04** | **User Setting Persistence:** Standard User changes a preference (e.g., theme, custom toolbar). Verify the setting is saved to %AppData% and persists across sessions. | Standard User |  |  |
| **A.5.05** | **App Updater Block:** Standard User attempts to run the in-app update/repair feature. **Must Fail** (or prompt for Admin). | Standard User |  |  |
| **A.5.06** | **Shortcut Behavior (Target):** Verify all shortcuts (Desktop/Start Menu) point to the correct executable path and icon. | Standard User |  |  |
| **A.5.07** | **Shortcut Removal:** Verify all shortcuts (All Users and Public Desktop) created during install are **removed** during silent uninstall. | Post Uninstall |  |  |
| **A.5.08** | **Uninstall Cleanliness:** Run the silent uninstall and confirm all files, folders, and registry keys created by the **package** are removed. | Post Uninstall |  |  |

## Tab 6: ☁️ Intune/CM Configuration

This tab details the specific settings required for successful deployment and monitoring via Microsoft Intune or Configuration Manager (CM).

| ID | Intune/CM Parameter | Configuration Detail | Status (Y/N) | Notes |
| --- | --- | --- | --- | --- |
| **I.6.01** | **Package Format:** Confirm the source files are wrapped correctly using the **Win32 Content Prep Tool** into a single .intunewin file. | File Check |  |  |
| **I.6.02** | **Install Command:** The command line entered into the portal (calling the PSADT wrapper). | powershell.exe -ExecutionPolicy Bypass -File .\Deploy-Application.ps1 -DeploymentType Install -DeployMode Silent |  |  |
| **I.6.03** | **Detection Method Type:** Select the most robust method (MSI Product Code, specific Registry Key, or custom PS script for complex apps). | MSI/Registry/Script |  |  |
| **I.6.04** | **Detection Rule Value:** The exact value/key/file version used to confirm *success* after installation. | e.g., `HKLM\SOFTWARE\App | DisplayVersion | >= 2.5.0` |
| **I.6.05** | **Supersedence:** Document if this package supersedes (replaces) an older version or a different package type. | Old App ID |  |  |
| **I.6.06** | **Requirements (Performance):** Verify Intune requirements meet minimum CPU speed, Disk Space, and RAM required by the vendor. | Built-in Rules |  |  |
| **I.6.07** | **Requirements (Custom):** Document any custom PowerShell requirements (e.g., checking for specific hardware models or prerequisite software). | Custom Script |  |  |
| **I.6.08** | **Device Restart Behavior:** Verify the Intune setting matches the PSADT handling (Determine behavior based on return codes or No specific action). | Setting Match |  |  |

## Tab 7: 🛡️ Security & Access Control

This tab focuses on hardening the application and ensuring compliance with the principle of least privilege.

| ID | Security Check | Standard/Goal | Status (Y/N) | Findings/Mitigation |
| --- | --- | --- | --- | --- |
| **S.7.01** | **File/Folder Permissions:** Verify the installed application directory grants **Modify** permissions only to **System** and **Administrators**, and **Read/Execute** to **Users**. | Least Privilege |  |  |
| **S.7.02** | **System Registry Access:** Verify the Standard User cannot modify settings in HKLM (which could break the application for other users). | Least Privilege |  |  |
| **S.7.03** | **Service Configuration:** Verify any installed service runs under a restricted account (e.g., Local Service or Network Service), not Local System, if possible. | Service Hardening |  |  |
| **S.7.04** | **Write to Program Files:** Verify the application does **not** attempt to write dynamic data, configuration, or log files to %ProgramFiles%. | Application Isolation |  |  |
| **S.7.05** | **Configuration Files:** Verify critical configuration/license files are protected in %ProgramData% or HKLM and are non-modifiable by Standard Users. | Data Protection |  |  |
| **S.7.06** | **Certificates:** Verify the package does not install expired or untrusted security certificates to the Trusted Root Store. | Certificate Audit |  |  |

## Tab 8: 💻 Environment & Performance (VDI/RDS/Performance)

This tab addresses considerations for virtualized or multi-user environments (VDI, RDS, Citrix) and general performance impact.

| ID | VDI/Performance Check | Test Environment | Status (Pass/Fail) | Impact/Mitigation |
| --- | --- | --- | --- | --- |
| **V.8.01** | **VDI/RDS Compatibility:** Install the app on a base image/template or gold master, and verify functionality in a multi-session environment. | VDI/RDS Test Bed |  |  |
| **V.8.02** | **Roamable Data:** Verify the application places all large, non-roamable data (e.g., cache, logs) into %LocalAppData% or a redirected path. | User Profile Analysis |  |  |
| **V.8.03** | **First Launch Speed:** Measure the application launch time on first use for a new user profile. | Stop Watch |  |  |
| **V.8.04** | **Logon Impact:** Verify the package installation does not add any scripts, GPOs, or entries that slow down the user logon process (especially critical in VDI). | Logon Simulator |  |  |
| **V.8.05** | **Disk Space Impact:** Document the exact disk space consumed by the package after installation. | File System Check |  |  |
| **V.8.06** | **Memory Footprint:** Monitor the application's memory usage during idle and peak load compared to baseline. | Performance Monitor |  |  |