
Slide 1: Title

(Opening) Hello everyone, my name is Krishna Verma, and today I'll be presenting my capstone project on adding a launch condition to an installer to restrict installation based on OS versions or hardware[1].

Slide 2: Introduction & What is a Launch Condition

(Introduction) My project focuses on **launch conditions**, which are rules set within an installer to ensure that software only runs on supported environments[2]. By adding conditions like operating system versions or specific hardware requirements, we can prevent installations on incompatible systems, which improves stability, security, and the overall user experience[3].

(What is Launch Condition) A launch condition is a rule that an installer checks before the installation begins[4]. If the computer doesn't meet the specified requirements, such as the correct Windows version or 64-bit hardware, the installation will stop and display a message[5].

Slide 3: Benefits & Prerequisites

(Benefits) Using launch conditions offers several key benefits:

- **Avoids Errors:** It prevents the installer from running on systems that don't meet the requirements, which reduces installation failures[6].
- **Stable Performance:** It ensures the software is installed only on compatible operating systems and hardware, leading to smoother performance[7].
- **Saves Time:** Both users and IT teams save time by avoiding unnecessary troubleshooting of failed installations[8].
- **Improves Security:** It blocks installations on outdated or unsupported systems that may pose security risks[9].

(Prerequisites) To complete this project, I needed a few things:

- Windows test machines or Virtual Machines (VMs) with different Windows versions[10].
- Virtualization software like Hyper-V, VirtualBox, or VMware to create these test environments[11].
- An installer authoring tool, specifically

Advanced Installer[12].

- Basic knowledge of MSI and installer principles[13].
- And administrator access to install and test the software in the VMs[14].

Slide 4: System Requirements & Workflow

(Minimum System Requirements) The system requirements for running the test environment and the tools are:

- **CPU:** A dual-core processor (2 GHz or higher) is needed for smooth VM operations[15].
- **RAM:** 4 GB or more ensures enough memory for the virtualization software and test OS environments[16].
- **Storage:** At least 40 GB of free space is required for tools, ISOs, and test files[17].
- **Display:** A resolution of 1366x768 or higher is needed for clear UI-based testing[18].
- **Operating System:** Windows 10 or later is necessary to install the required tools[19].

(Workflow) I followed a clear five-step process:

1. Start with a fresh installer in Advanced Repackager on a clean VM[20].
2. Unpack the installer and move it to Advanced Installer for modification[21].
3. Add and modify the launch condition[22].
4. Build the new installer[23].
5. Install it on a test VM and collect the logs[24].

Slide 5: Step-by-Step Process & Results

(Steps) The process began by creating a folder for the repackaged MSI file[25]. I then used the application repackaging tool to "Capture Setup," where I selected the MSI file and gave a name for the output[26]. After the capture process, I uninstalled any existing versions of the application[27].

Next, I opened the package in

Advanced Installer and navigated to the **Launch Conditions** section[28]. I deselected the "Any Windows Version" option and chose the specific OS versions I wanted to allow, such as Windows Server 2022 x64[29]. After that, I went to

Product Details and clicked **Build** to save the final MSI file[30303030].

(Result) To test the launch condition, I went to the **Launch Conditions** section again and selected a random Windows version that was not on my current system[31]. I then rebuilt the MSI file. When I tried to install this newly built MSI, the installation failed, which confirmed that our launch condition was successful[32].

Slide 6: Conclusion

(Conclusion) After completing this project, I successfully implemented a launch condition that restricts software installation based on specific OS versions and hardware[33]. This project ensures compatibility, prevents installations on unsupported systems, and improves overall deployment reliability[34].

Slide 7: Thank You

(Closing) Thank you for your time. Are there any questions?