

## **WIPRO Capstone Project-2**

**Project-Name:** Customize MSIX App Launch Behavior - Modify the AppxManifest.xml to control app behavior and appearance.

**Prepared By:** 28953 - Kranti Jadhav

**User-ID:** 28953

**Training Batch:** Wipro\_25VID1334\_DWS - Batch 1

**Project-Batch No:** 12

## **1. About Project**

This project helps us understand how to modify an MSIX app's behavior during launch using the AppxManifest.xml file.

In this project, I open and edit the AppxManifest.xml to change how our app behaves when it is launched. The manifest file is a key part of every MSIX package, and it controls how the app looks and functions when it is launched. It includes details like the app name, icon, description, background color, entry point, and other settings. We can manually extract the MSIX file and edit the AppxManifest.xml file to customize the app's display name, description, icons, and colors. Then repack and sign it again so that the app behaves as we want it to.

## **2. Prerequisites**

- MSIX Packaging Tool (from Microsoft Store)
- A .msix package
- A valid .pfx certificate Visual Studio Code or
- Notepad for editing XML files

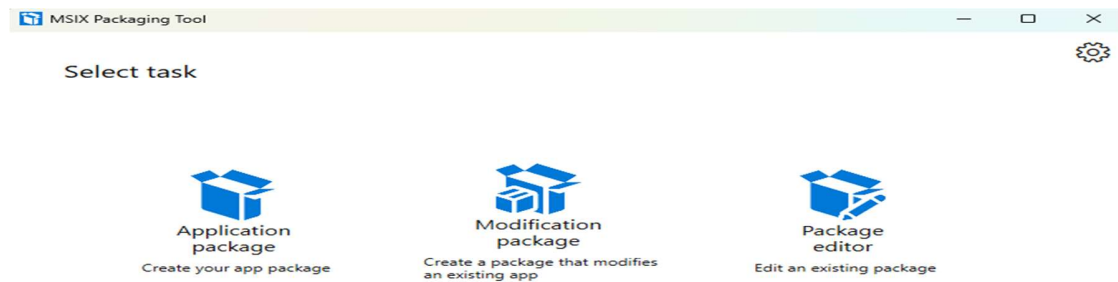
## **3. Tools and Technologies Used**

- MSIX Packaging Tool
- PowerShell
- Notepad++

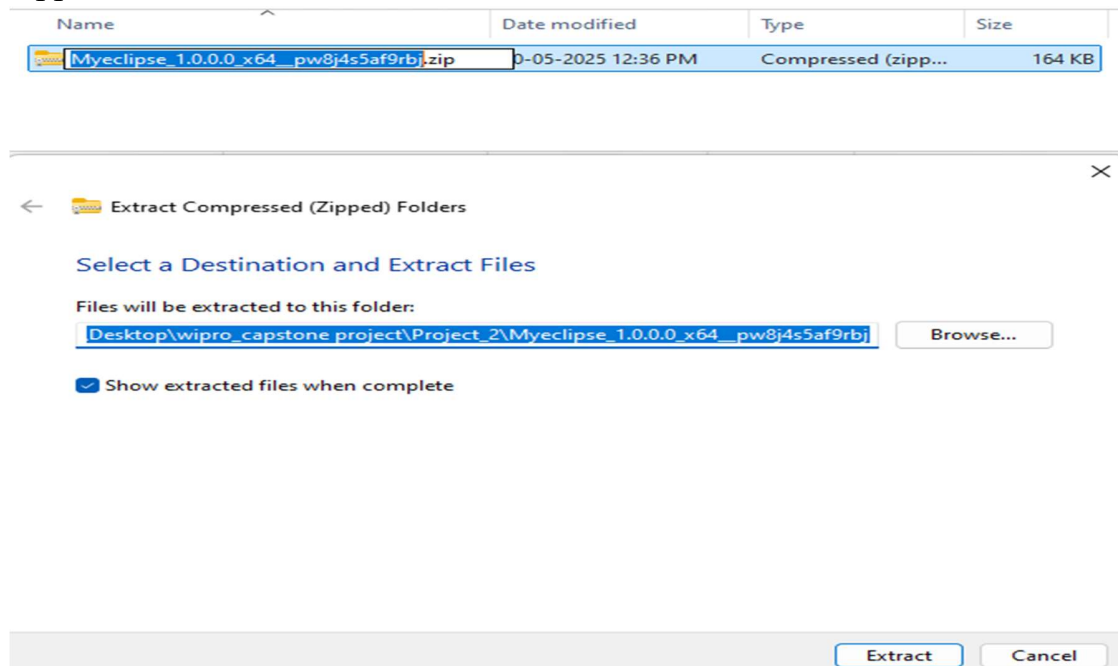
## 4. Tasks Performed

### Steps to Customize MSIX App Launch Behavior:

**Step 1:** Create or Open an MSIX Package use the MSIX Packaging Tool to create your MSIX package. Or use an existing MSIX package that you already built



**Step 2:** Extract the MSIX Package - Rename your .msix file to .zip Right-click and **Extract All** to see all its internal files. You will find a file named: AppxManifest.xml



**Step 3:** Open AppxManifest.xml using Notepad++, Visual Studio Code, or even simple Notepad.

Assets	16-06-2025 11:26 PM	File folder	
VFS	16-06-2025 11:26 PM	File folder	
[Content_Types].xml	30-05-2025 12:36 PM	XML File	1 KB
AppxBlockMap.xml	30-05-2025 12:36 PM	XML File	2 KB
AppxManifest.xml	30-05-2025 12:36 PM	XML File	2 KB
Registry.dat	30-05-2025 12:36 PM	DAT	32 KB
Resources.pri	30-05-2025 12:36 PM	PRI File	1 KB
User.dat	30-05-2025 12:36 PM	DAT	24 KB
UserClasses.dat	30-05-2025 12:36 PM	DAT	12 KB

**Step 4:** Make Custom Changes in the file such as DisplayName, Description, Logo etc.

```
File Edit View

<?xml version="1.0" encoding="utf-8"?>
<Package xmlns="http://schemas.microsoft.com/appx/manifest/foundation/windows10" xmlns:uap10
="http://schemas.microsoft.com/appx/manifest/uap/windows10/10"
xmlns:rescap="http://schemas.microsoft.com/appx/manifest/foundation/windows10/restrictedcapabilities"
IgnorableNamespaces="uap10 rescap">
  <!--Package created by MSIX Packaging Tool version: 1.2024.405.0-->
  <Identity Name="Myeclipse" Publisher="CN=Myeclipse" Version="1.0.0.0" ProcessorArchitecture="x64" />
  <Properties>
    <DisplayName>Myeclipsepack</DisplayName>
    <PublisherDisplayName>Myeclipsepackage</PublisherDisplayName>
    <Description>None</Description>
    <Logo>Assets\StoreLogo.png</Logo>
    <uap10:PackageIntegrity>
      <uap10:Content Enforcement="on" />
    </uap10:PackageIntegrity>
  </Properties>
  <Resources>
    <Resource Language="en-us" />
  </Resources>
  <Dependencies>
    <TargetDeviceFamily Name="Windows.Desktop" MinVersion="10.0.17763.0" MaxVersionTested="10.0.22000.1" />
  </Dependencies>
</Package>
```

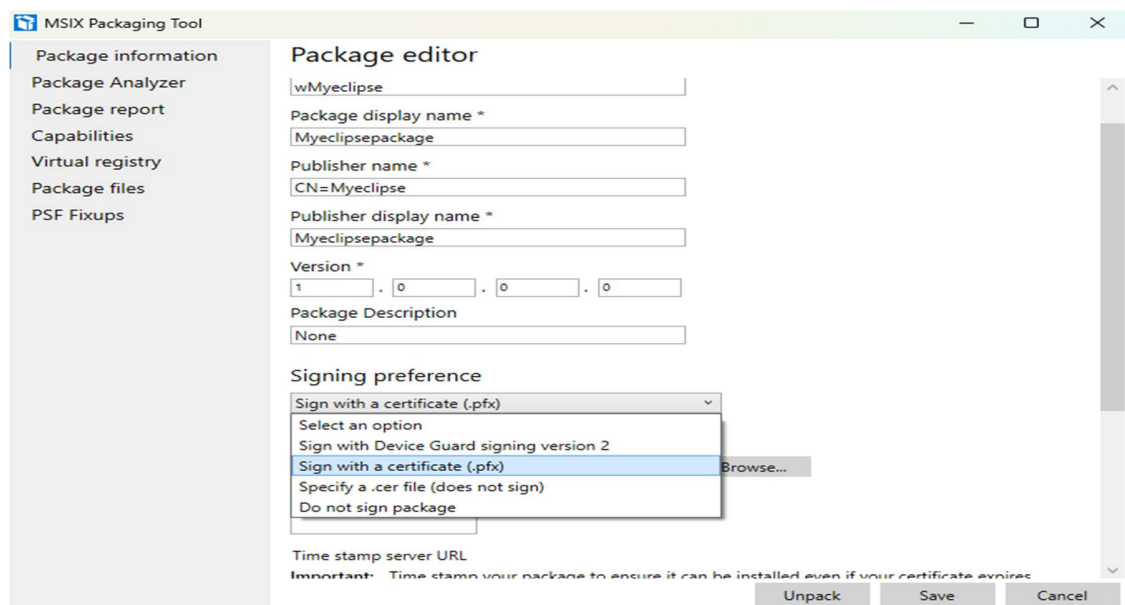
Here, I have changed the DisplayName and the Description

```
<!--Package created by MSIX Packaging Tool version: 1.2024.405.0-->
<Identity Name="Myeclipse" Publisher="CN=Myeclipse" Version="1.0.0.0" ProcessorArchitecture="x64" />
<Properties>
  <DisplayName>Myeclipsepack</DisplayName>
  <PublisherDisplayName>Myeclipsepackage</PublisherDisplayName>
  <Description>Changed the Description for Myeclipseapp</Description>
  <Logo>Assets\StoreLogo.png</Logo>
  <uap10:PackageIntegrity>
    <uap10:Content Enforcement="on" />
  </uap10:PackageIntegrity>
</Properties>
```

**Step 5:** After editing the XML, **save** the file. Select all extracted files including the updated AppxManifest.xml. Zip them again. Rename .zip to .msix.

Myeclipse_1.0.0.0_x64_pw8j4s5af9rbj	16-06-2025 11:41 PM	File folder	
Myeclipse_1.0.0.0_x64_pw8j4s5af9rbj.msix	06-2025 11:42 PM	MSIX File	159 KB

**Step 6:** use **MSIX Packaging Tool** or **PowerShell** to sign the updated .msix file using your certificate.



## 5. Analysis Output

I was able to successfully customize the launch behavior of an MSIX app by modifying its AppxManifest.xml file. This included changing the app name, icon, background color, and more. I also learned how to repackage and re-sign the MSIX file properly. This gave me a deeper understanding of how MSIX works and how to control app behavior visually.

