extension requires adding a reference to a WiX UI extension package and configuring the `WixVariable` elements in your `wxs` file. The standard MSI Package template does not include a complex user interface by default.

Step-by-Step Guide

The following steps outline how to add a standard user interface (including a welcome page) to a WiX v4 installer project in Visual Studio using the HeatWave extension.

1. **Install the HeatWave Extension**
   Ensure you have the FireGiant HeatWave extension installed for your version of Visual Studio (2019 or 2022) from the Visual Studio Marketplace.

2. **Add the WiX UI Extension NuGet Package**
   The default `MSI Package` project template creates an installer that uses the basic Windows Installer UI. To add a modern, customizable UI (like the classic "welcome" -> "license" -> "install" flow), you must add a NuGet package reference:

   1. Right-click your WiX installation project in the **Solution Explorer**.
   2. Select **Manage NuGet Packages**.
   3. In the **Browse** tab, search for `WixToolset.UI.wixext`.
   4. Install the latest version of this package.

3. **Configure the WiX Source File (.wxs)**
   Modify your main `wxs` file to use the new UI extension. This involves adding the necessary XML namespace and referencing the standard UI dialog sets.

   1. Open your main `.wxs` file (e.g., `Package.wxs`).
   2. Add the `xmlns:ui` namespace to the root `<Wix>` element:

      xml

      ```
      <Wix xmlns="http://wixtoolset.org/schemas/v4/wxs" xmlns:ui="http://wixtoolset.
      ```

   3. Inside the `<Product>` or `<Package>` element, add the `<ui:WixUI>` element and specify the desired dialog set. The `WixUI_InstallDir` is a common choice as it includes a welcome page, a license agreement page, an installation directory

xml

```xml
<Product ...>
    ...
    <ui:WixUI Id="WixUI_InstallDir" />
    ...
</Product>
```

4. **Add a License File (if using `WixUI_InstallDir`)**
   The `WixUI_InstallDir` dialog set expects a `License.rtf` file in your project directory.
   1. Add a Rich Text Format (`.rtf`) license file to your project.
   2. To make it recognized by the UI, you set a `WixVariable` in your `wxs` file, usually before the `<ui:WixUI>` element:

   xml

   ```xml
   <WixVariable Id="WixUILicenseRtf" Value="License.rtf" />
   ```

5. **Build the Project**
   Build your WiX project in Visual Studio. The resulting MSI installer will now include the new user interface with the welcome page and subsequent dialogs.

Example `Package.wxs` Structure

xml

```xml
<Wix xmlns="http://wixtoolset.org/schemas/v4/wxs" xmlns:ui="http://wixtoolset.org">
    <Package ...>
        ...
        <!-- Define the license file for the UI -->
        <WixVariable Id="WixUILicenseRtf" Value="License.rtf" />

        <!-- Reference the standard InstallDir UI dialog set (includes a welcome pa
        <ui:WixUI Id="WixUI_InstallDir" />
```

```
            ...
        </Feature>
    </Package>
</Wix>
```

Adding the WiX extension to our project | FireGiant Docs

Adding a WiX extension to a WiX project * Right-click on WixTutorialPackage . * Choose Manage NuGet Packages . * Click Browse in t...

FireGiant Docs

How to Add UI to installer for wix v4 (HeatWave for VS2022 ...

Sep 13, 2023 — Related * 2925. * 540. Error 'LINK : fatal error LNK1123: failure during conversion to COFF: file invalid or corrupt' after instal...

Stack Overflow

Install HeatWave | FireGiant Docs

We're building our app and its installer in Visual Studio, so we need a Visual Studio extension that knows all about WiX. Luckily,

FireGiant Docs

Show all

Dive deeper in AI Mode

Videos

31:51

HeatWave! The New Hotness for WiX v4

YouTube · Rob Mensching