



# Quick Start Manual

Optix PC Screen Capture

OPTIXPC-SCREENCAPTURE-QS001-EN-AUG-2025

## Contents

End-User License Agreement (EULA) .....	2
Third-Party Trademarks Disclaimer .....	4
Introduction .....	5
Important User Information .....	6
Prerequisites .....	7
System used while testing - Hardware & Software .....	7
Sharadsoft software details .....	7
Screen capture in Optix PC .....	8
Configuration-MainWindow, Screens, Tags, RuntimeNetLogic.....	8

## End-User License Agreement (EULA)

End-User License Agreement (EULA) for Software developed by Sharadsoft

Last Updated: 01-Mar-2025

### 1. Introduction

This End-User License Agreement (EULA) is a legal agreement between you (either an individual or a single entity) and Sharadsoft for the Software developed by Sharadsoft, which includes computer software and may include associated media, printed materials, and "online" or electronic documentation ("Software").

### 2. License Grant

Sharadsoft grants you a revocable, non-exclusive, non-transferable, limited license to download, install, and use the Software solely for your personal, non-commercial, commercial purposes strictly in accordance with the terms of this Agreement. Any software offered for free by Sharadsoft must be provided free of charge to sub-customers, and license/activation does not apply to such free Software.

### 3. Restrictions

You agree not to, and you will not permit others to:

- Decompile, reverse engineer, disassemble, attempt to derive the source code of, or decrypt the Software.
- Make any modification, adaptation, improvement, enhancement, or translation work from the Software.
- Violate any applicable laws, rules, or regulations in connection with your access or use of the Software.
- Remove, alter, or obscure any proprietary notice (including any notice of copyright or trademark) of Sharadsoft or its affiliates, partners, suppliers, or the licensors of the Software.

### 4. Termination

This Agreement is effective from the date you first use the Software and shall continue until terminated. You may terminate this Agreement at any time by uninstalling and deleting the Software and all copies thereof. This Agreement will terminate immediately if you fail to comply with any of its Terms.

### 5. Intellectual Property

All titles, including but not limited to copyrights, in and to the Software and any copies thereof are owned by Sharadsoft or its suppliers. All rights not expressly granted are reserved by Sharadsoft.

### 6. Disclaimer of Warranties

The Software is provided "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

### 7. Limitation of Liability

In no event shall Sharadsoft be liable for any special, incidental, indirect, or consequential damages whatsoever (including, but not limited to, damages for loss of profits, loss of data, or other information, for business interruption, for personal injury, for loss of privacy arising out of or in any way related to the use of or inability to use the Software, third-party Software and/or third-party hardware used with the Software, or otherwise in connection with any provision of this Agreement), even if Sharadsoft has been advised of the possibility of such damages and even if the remedy fails of its essential purpose.

#### 8. Governing Law

This Agreement shall be governed by and construed in accordance with the laws of Mumbai-Maharashtra-India, without regard to its conflict of law principles.

#### 9. Entire Agreement

This Agreement constitutes the entire Agreement between you and Sharadsoft with respect to the use of the Software and supersedes all prior or contemporaneous understandings regarding such subject matter.

## Third-Party Trademarks Disclaimer

### Disclaimer:

All product names, logos, and brands are the property of their respective owners. All company, product, and service names used in this manual are for identification purposes only. Use of these names, logos, and brands does not imply endorsement.

Sharadsoft acknowledges all third-party trademarks and logos referenced in this manual, including but not limited to:

- FactoryTalk Optix Studio is the trademarks of Rockwell Automation.

Any third-party trademarks or service marks referenced in this manual are the property of their respective owners. The inclusion of any third-party trademarks does not imply a partnership or endorsement by Sharadsoft.

For any trademark-related concerns or inquiries, please contact us at [info@sharadsoft.com](mailto:info@sharadsoft.com)

## Introduction

We are happy to deploy the library for Optix Studio, developed with a focus on fulfilling your specific screen capture requirements at the PC level, running on the Windows platform. While this library offers the essential functionality you need, please note that it is designed to provide a simple and user-friendly experience to achieve the desired outcomes of screen capture at the Optix PC level.

### Key Highlights:

1. **Simplicity and Efficiency:** Our library is built to offer a straightforward and efficient way to get the screen capture functionality at the Optix PC level without the complexity of the Optix NetLogic development.
2. **Tailored to Your Needs:** We have prioritized the features that are most relevant to your screen capture requirements, ensuring that the tool meets your specific needs without overwhelming you with unnecessary options.
3. **Ease of Use:** Allows for quick and easy integration, making it accessible even for users with minimal technical expertise.
4. **Focused Functionality:** While it may not have all the advanced features of professional parameter configuration and feedback mechanisms, our tool is enough to handle your screen capture tasks effectively.

We understand that you may compare our library with more comprehensive professional alternatives. It's important to recognize that our library is designed to provide the essential functionality required for your screen capture needs, focusing on simplicity and ease of use. This approach allows us to deliver a solution that is both cost-effective and user-friendly.

## Important User Information

- The points and notes mentioned in this manual are based on our sole understanding and the data available to us as of Aug 1, 2025. Please be aware that interpretations or meanings may vary. Kindly refer to the respective third-party manuals or documentation for further clarification or detailed information.
- The WinScreenCaptureLibrary library is compatible with Windows-based PCs only.
- Ensure the path where the screenshots are saved is appropriately configured. It is preferable to create a saved path folder before running the Optix runtime.
- Images used in the demo application are for demo purposes only and licensed by Sharadsoft for their use in an application developed by Sharadsoft. To maintain optimal CPU performance, avoid using multiple images within the application. Excessive image usage can lead to increased processing demands, potentially impacting overall application efficiency.

## Prerequisites

### System used while testing - Hardware & Software

Description	Details
Demo App Development Software	FT Optix Studio v1.6.2
Windows Operating System	Windows 11 Pro
Visual Studio	2022
Workstation RAM	8 GB or more
Sharadsoft Software	1.00.00

### Sharadsoft software details

Description	Activation	Functionality
WinScreenCaptureLibrary	Requires *1,2	Screen capture during Optix runtime on Windows PC

**Note:**

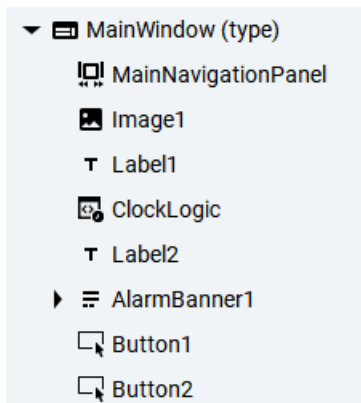
1. WinScreenCaptureLibrary is a separate purchase and requires an individual activation key.
2. Without activation, it will take a screenshot with diagonal lines overlapped.

## Screen capture in Optix PC

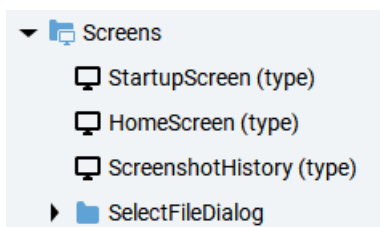
Your first step will be creating the Optix application for PC using Optix Studio.

[Configuration-MainWindow](#), [Screens](#), [Tags](#), [RuntimeNetLogic](#)

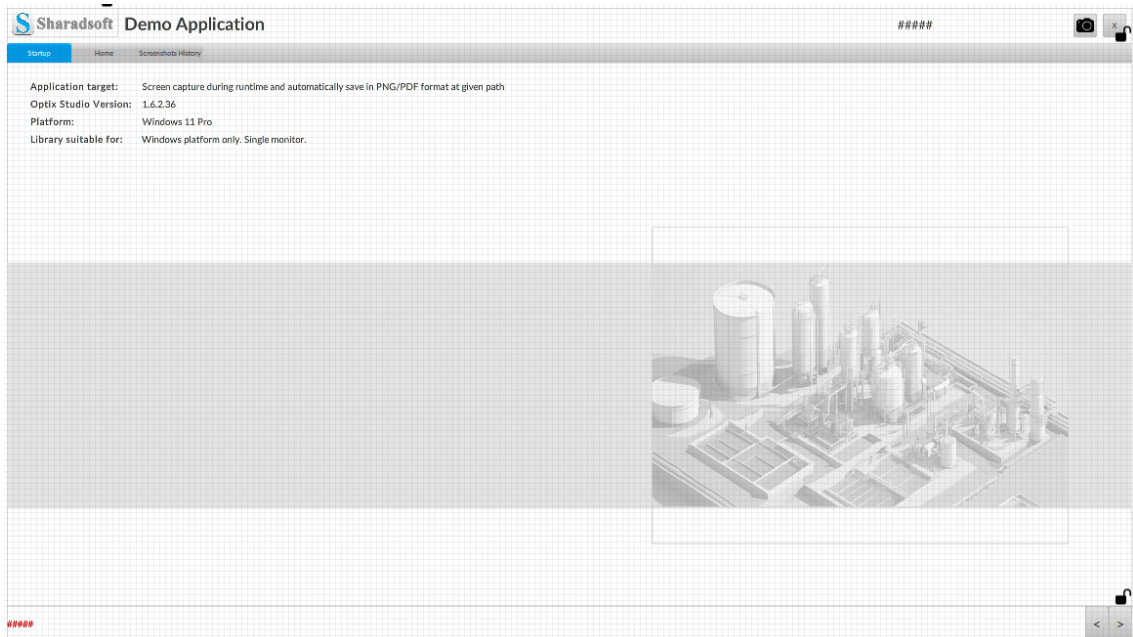
1. Configure the 'MainWindow' section with your required screen navigation option. For the demo application, we have used a navigation panel as follows:



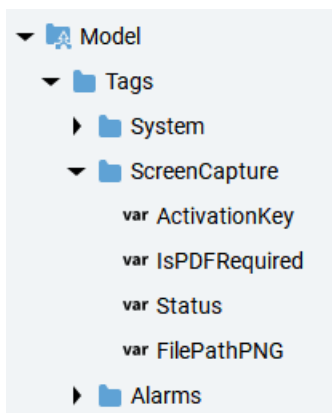
2. Create and configure the required screens for your application. For the demo application, we have created and configured 'StartupScreen', 'HomeScreen', and 'ScreenshotHistory' named screens and added them to the main window navigation panel.



3. MainWindow is designed for common things like app close button, screenshot button, DateTime label at the top side of MainWindow, and Alarm Banner at the bottom.



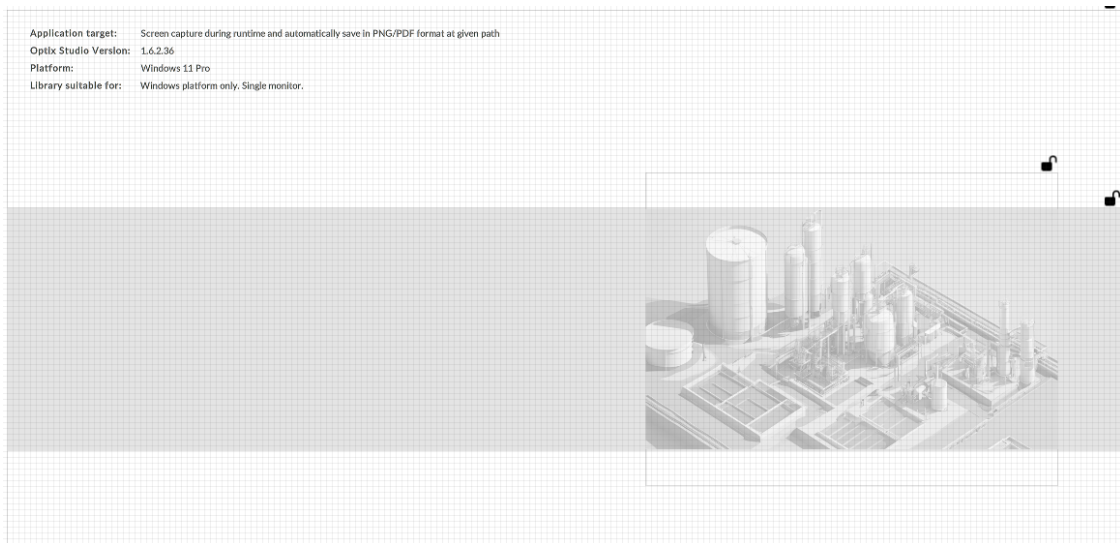
4. Create the tags under the 'Model' section. For the demo application, we have created the tags as follows:



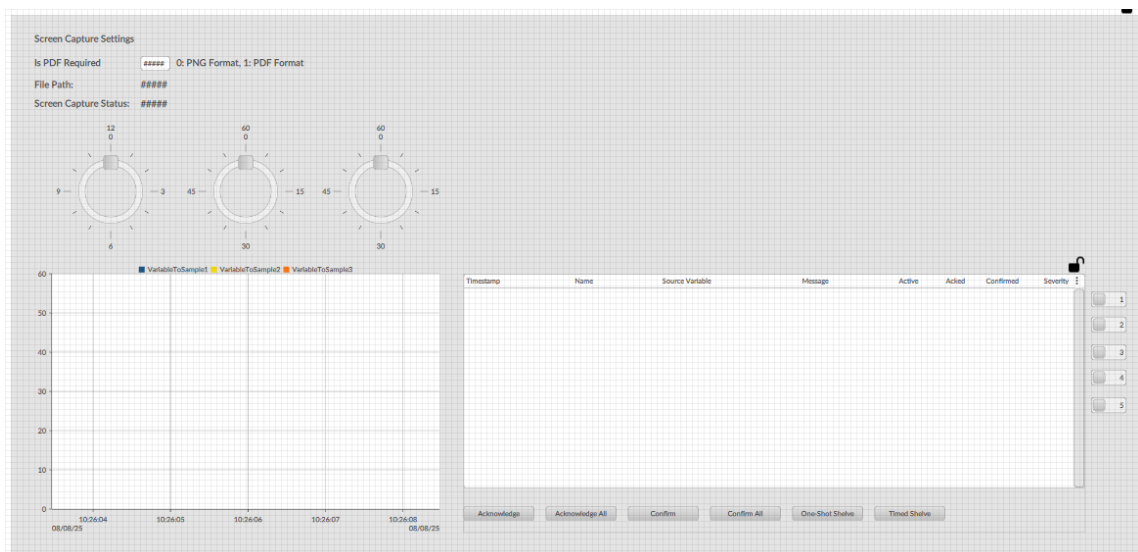
ActivationKey	String	AAAA-BBBB-CCCC-DDDD-EEEE-FFFF-1111-2222-3333-4444-5555
IsPDFRequired	Int32	0
Status	String	0
FilePathPNG	String	0

You can set the 'ActivationKey' tag value here only. 'IsPDFRequired' tag value can be fixed here as 0 for PNG format screenshot or 1 for PDF format screenshot. 'Status' tag is the read-only tag that will give you the screen capture command execution status. 'FilePathPNG' is the tag where you need to pass the screenshot file path during runtime.

5. The startup screen developed for the demo application is as follows:

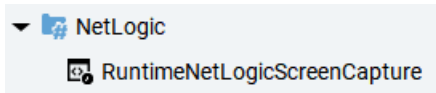


6. The home screen developed for the demo application is as follows:

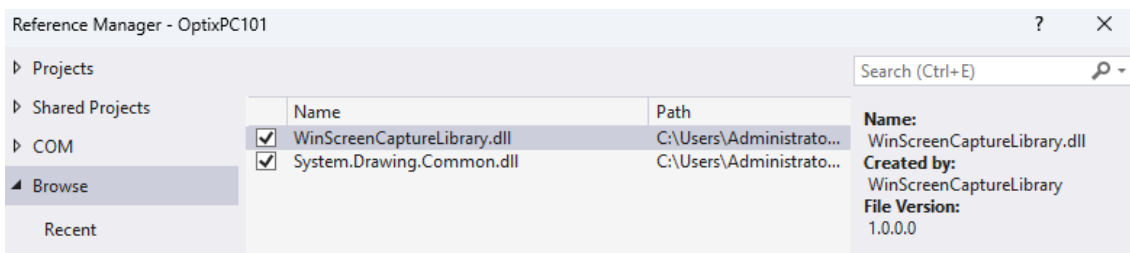
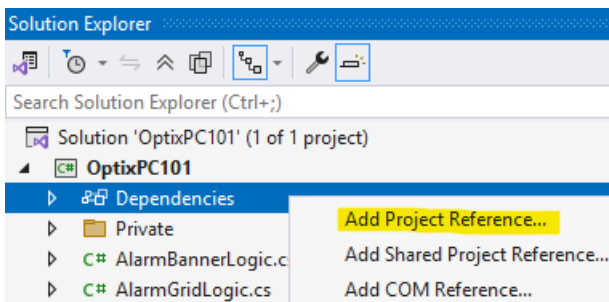
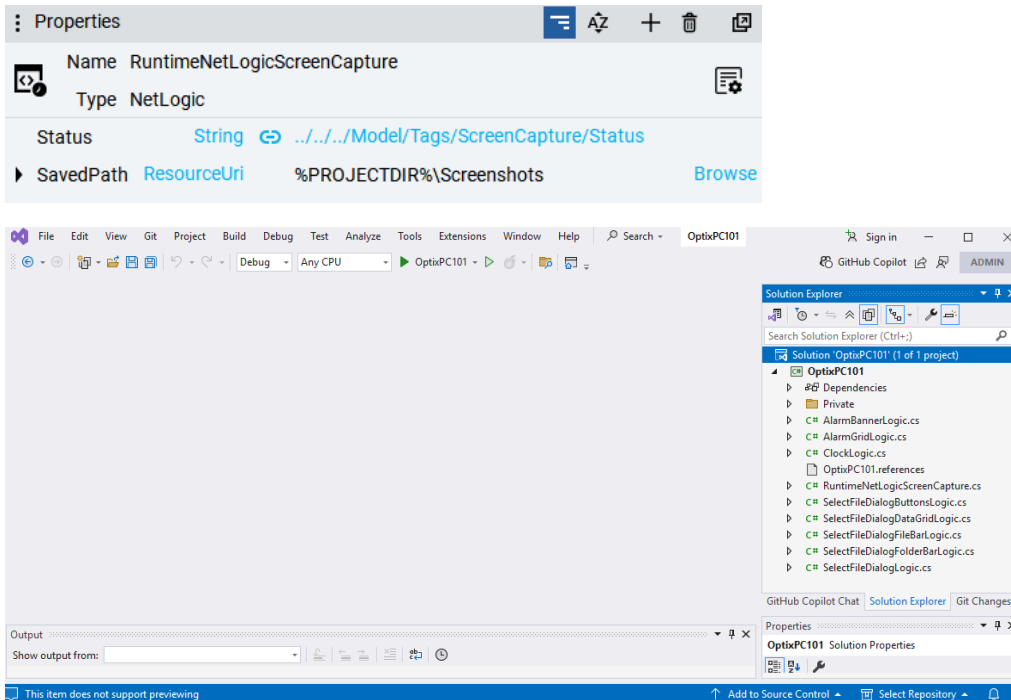


This screen is developed for the screen capture demonstration. This screen included live data trend & circular gauges, and an alarm grid to check the graphic quality in the screenshot.

7. Add the 'RuntimeNetLogicScreenCapture' script under 'NetLogic' section. Open the created NetLogic using Visual Studio 2022 for editing purposes. Add the 'WinScreenCaptureLibrary' & 'System.Drawing.Common' dll files as a reference in the dependencies section. Edit/copy the script as per the given demo application script.



Set the properties of the NetLogic as follows:



```

> Using directives
}

public class RuntimeNetLogicScreenCapture : BaseNetLogic
{
    WinScreenCapture MyWinScreenCapture = new WinScreenCapture();

    public override void Start()
    {
        // Insert code to be executed when the user-defined logic is started
    }

    public override void Stop()
    {
        // Insert code to be executed when the user-defined logic is stopped
    }

    [ExportMethod]

    public void CaptureScreen(string ActivationKey, Int32 IsPDFRequired)
    {
        string Status = string.Empty;

        LogicObject.GetVariable("Status").Value = Status;

        string MyFilePath= GetFilePath();
        Log.Info(MyFilePath);

        MyWinScreenCapture.CaptureScreen(ActivationKey, MyFilePath, IsPDFRequired, out Status);

        LogicObject.GetVariable("Status").Value = Status;
    }

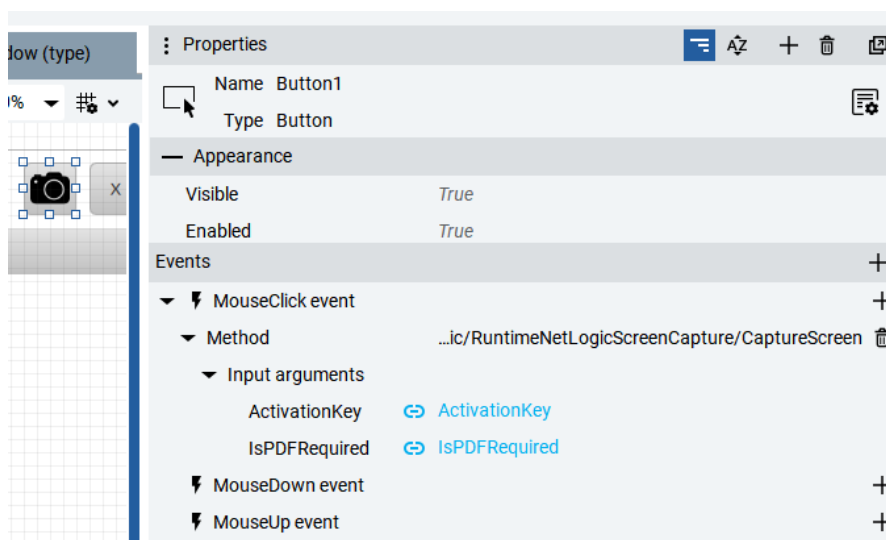
    private string GetFilePath()
    {
        var PathVariable = LogicObject.GetVariable("SavedPath");
        if (PathVariable == null)
            throw new ArgumentException("SavedPath variable not found");

        string Path = LogicObject.GetVariable("SavedPath").Value;
        if (string.IsNullOrEmpty(Path))
            throw new ArgumentException("SavedPath path is empty");

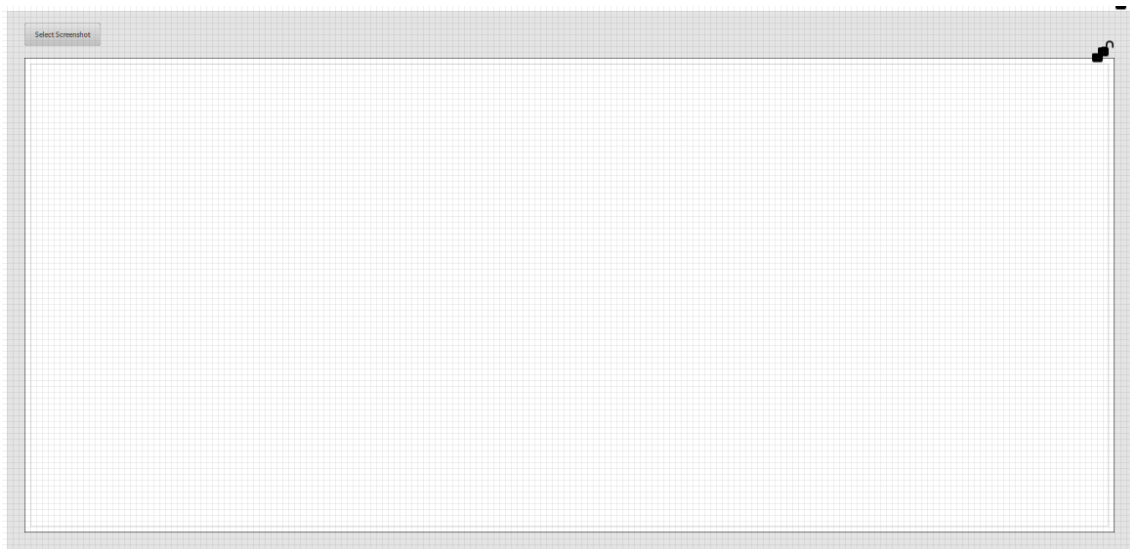
        return new ResourceUri(PathVariable.Value).Uri;
    }
}

```

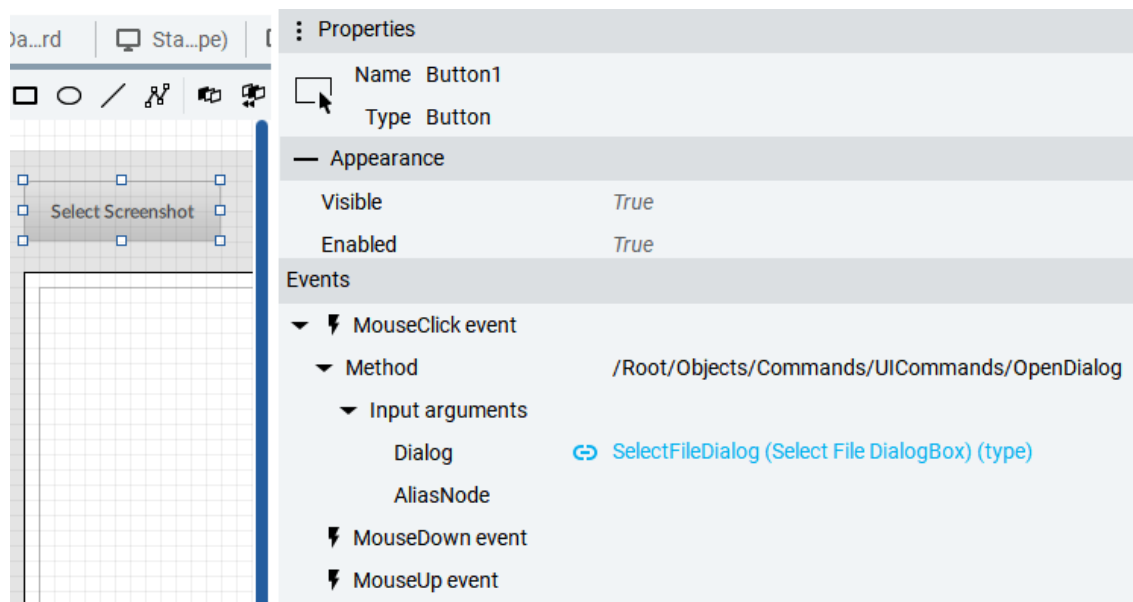
8. 'Screenshot' button event configuration is as follows:



9. The ScreenshotHistory screen developed for the demo application is as follows:

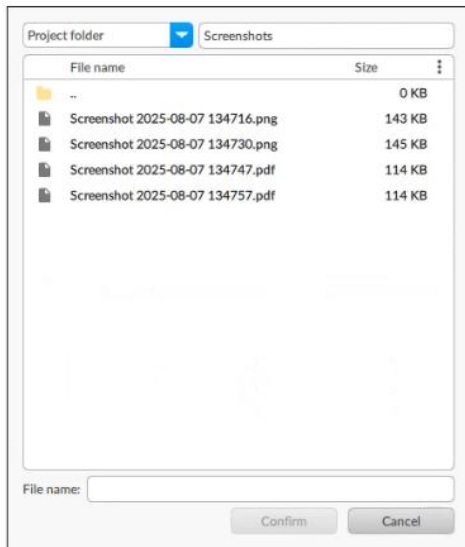


'Select Screenshot' button properties is as follows:

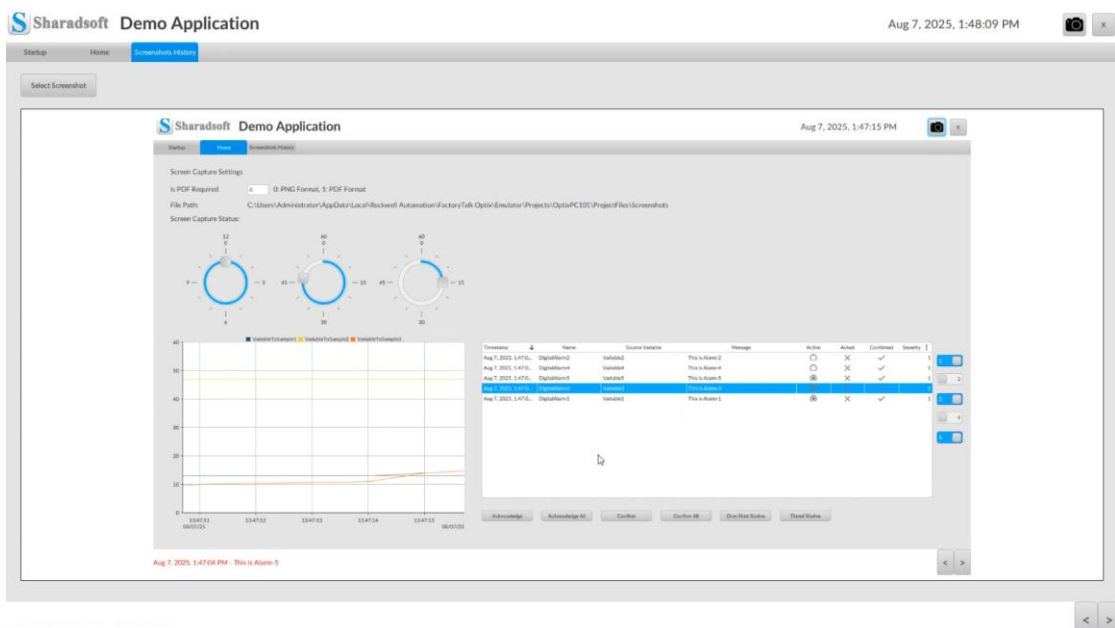
A screenshot of a software development environment showing the properties panel for a button. The panel is titled "Properties" and lists various attributes for the button, including its name, type, appearance, and events. The button is named "Button1" and is of type "Button". Its appearance is visible and enabled. The events list includes "MouseClicked event", "MouseDown event", and "MouseUp event". The "MouseClicked event" is expanded to show its method, "OpenDialog", and its input arguments, including a "Dialog" property set to "SelectFileDialog (Select File DialogBox) (type)".

Properties	
Name	Button1
Type	Button
Appearance	
Visible	True
Enabled	True
Events	
MouseClicked event	
Method	/Root/Objects/Commands/UICommands/OpenDialog
Input arguments	
Dialog	SelectFileDialog (Select File DialogBox) (type)
AliasNode	
MouseDown event	
MouseUp event	

During runtime, 'Select Screenshot' button will call 'Select File' window as shown below:



Selected screenshot will be displayed as shown in the following snap:



10. Now you are ready with the application for testing. Close Visual Studio if it is open. Save the application and run the runtime.

[www.sharadsoft.com](http://www.sharadsoft.com)

---

[Feedback/Support](#)

Your comments will help us better serve your Optix PC level screen capture needs. If you have any suggestions/ require support, then drop an email to 'info@sharadsoft.com'.

Location: Mumbai, India.

Document updated on 08-Aug-2025