



UWP



Electronic signatures



WINDOWS > GUIDES > SIGNATURES > ELECTRONIC SIGNATURES

Add electronic signatures to PDFs in Windows

This guide explains how to add an electronic signature (eSignature) to a PDF document on Windows using Nutrient UWP SDK, both through the built-in user interface (UI) and programmatically.

Adding an electronic signature programmatically

In Nutrient UWP SDK, electronic signatures are implemented as PDF annotations, commonly referred to as signature annotations.

Electronic signatures can be either ink or image annotations, created with:

- ❖ `PSPDFKit.Pdf.Annotation.Ink` for drawn signatures.
- ❖ `PSPDFKit.Pdf.Annotation.Image` for image-based signatures.

To add an eSignature:

- 1 Set the `IsSignature` property to `true`.
- 2 Add the annotation using `CreateAnnotationAsync`.

Licensing requirements for signature annotations

Creating, updating, and deleting signature annotations requires a license that includes either the **Annotations** component or the **Electronic Signatures** component. If your license includes only the **Electronic Signatures** component (without the **Annotations** component), modifications are restricted



ASK AI

to signature annotations exclusively. Any attempt to modify non-signature ink or image annotations (`isSignature = false`) or other annotation types will be disallowed.

Creating an ink signature

To create an ink signature programmatically, use to the following code snippet:

```
1 var annotation = new Ink
2 {
3     PageIndex = 0,
4     IsSignature = true,
5     Lines = new List<ILine<DrawingPoint>>
6     {
7         new List<DrawingPoint>
8         {
9             new DrawingPoint(x: 5, y: 5),
10            new DrawingPoint(x: 95, y: 95)
11        },
12        new List<DrawingPoint>
13        {
14            new DrawingPoint(x: 95, y: 5),
15            new DrawingPoint(x: 5, y: 95)
16        }
17    },
18    BoundingBox = new Rect(
19        x: 0,
20        y: 0,
21        width: 100,
22        height: 100
23    )
24 };
25
26 await PdfView.Controller.CreateAnnotationAsync(annotation);
```

Creating an image signature

To create an image signature programmatically, use to the following code snippet:

```
1 // Read the image in a buffer.
2 var buffer = await FileIO.ReadBufferAsync(file);
3
4 // Create an annotation attachment for the image in the document.
5 var attachment = await PdfView.Document.CreateAttachmentAsync(buffer, file.Con
6
7 // Create the annotation template and set the image attachment ID and the cont
```

```

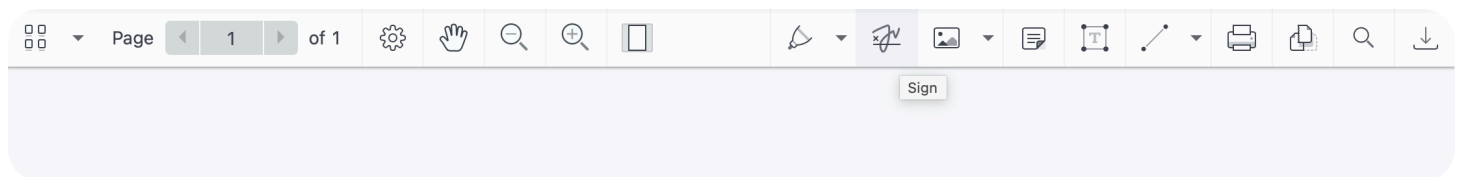
8  var annotation = new Image
9  {
10     PageIndex = 0
11     IsSignature = true,
12     // In this case, "image/png".
13     ContentType = file.ContentType,
14     ImageAttachmentId = attachment,
15     Description = "Signature Image",
16     BoundingBox = new Rect(100, 100, 100, 100),
17 };
18
19 await PDFView.Controller.CreateAnnotationAsync(annotation);

```

Using the built-in UI

If you're using the SDK with [Forms](#), end users can initiate the signature creation modal by tapping a signature form field within the document. If no signature form field is present, end users can manually add a signature using the signature tool button.

The signature tool button is available in the toolbar for quick access.



Signature creation modal view

When the signature creation modal view is displayed, end users can add a signature using one of three methods:

- ✧ **Draw** — End users can create a handwritten signature using a touchscreen or stylus. This method is particularly effective when using devices such as Surface Pen.
- ✧ **Attach an image** — End users can attach an existing signature image from their device.
- ✧ **Type** — End users can enter their signature as text using a chosen font style.



End users can attach an existing signature image from their device. If the hardware supports it, they can also capture a photo of their handwritten signature on paper for a digital scan. This option is ideal when signing on a device with easy access to stored files.



End users can enter their name and select from a predefined set of font-based signature styles. This method ensures accessibility and is fully compatible with:

- ⌘ Screen readers such as VoiceOver, TalkBack, NVDA, and JAWS.
- ⌘ Assistive technologies such as Switch Control on Mac and iOS.

By default, Nutrient provides four signature styles, and you can customize the available options by defining a list of preferred fonts.



Color options

For both the **Draw** and **Type** options, end users can choose between black and two shades of blue to ensure the signature remains distinguishable from the document background.

Was this helpful?

☒ YES

☐ NO

