Create missing custom renderer for single platform

Asked 4 months ago Active 2 days ago Viewed 95 times

1

The bounty expires in 2 days. Answers to this question are eligible for a +50 reputation bounty. raV720 wants to **draw more attention** to this question.

I need to use PdfViewer from some library. This library provides control for Android, IOS, UWP platforms but not for WPF platform. How to create a renderer only for WPF platform?

WPF renderer:

```
[assembly: ExportRenderer(typeof(PdfViewer), typeof(PdfViewerRenderer))]
public class PdfViewerRenderer : Xamarin.Forms.Platform.WPF.ViewRenderer<PdfViewer,
WPF.PlatformImplementations.PdfViewer>
{

}
```

WPF control implementation:

```
namespace WPF.PlatformImplementations
{
    public class PdfViewer
    {
     //custom implementation of pdf viewer for wpf
    }
}
```

Net standard library contains PdfViewer which derives from SfPdfViewer from a library:

```
namespace Views
{
    public class PdfViewer : SfPdfViewer
    {
     //implementation of pdf viewer using base class from multiplatform library
    }
}
```

The above approach does not work for WPF and works for other platforms. When I change inheritance from SfPdfViewer to Xamarin.Forms.View then it works for WPF but obviously does not work for other platforms. So the problem is to make renderer mechanizm to recognize Views.PdfViewer as control based on Xamarin.Forms.View.

c# xamarin xamarin.forms Edit tags

edited Apr 7 at 14:08

asked Nov 28 '19 at 10:36
raV720

1 One way to do this is to create a `CustomPdfViewer` interface in the Core project, and have individual implementations in each native project using Dependency injection. In the native implementations for each of the iOS, Android, & UWP projects, just make it use the PdfViewer library that works for you. And then for WPF, just use the SfPdfViewer instead. – Saamer Apr 7 at 22:08

@Saamer approach seems correct – Ricardo Dias Morais Apr 7 at 22:25

If you have purchased their subscription they will probably add support for WPF if you request – Prateek 10 hours ago

1 Answer

Active | Oldest | Votes

0

One way to do this is to create a `CustomPdfViewer` interface in the Core project, and have individual implementations in each native project using Dependency injection.

In the native implementations for each of the iOS, Android, & UWP projects, just make it use the `PdfViewer` library that works for you.

And then for WPF, just use the `SfPdfViewer` instead.

Eg: You can take a look at the official documentation here on DependencyService with examples

Once you do this, you will have your solution. But if you want to go one step further and access the `CustomPdfViewer` you created inside your XAML, you can use "XAML Markup Extensions" represented by the term `local` as you can also see in the example here.

edited 2 days ago

answered Apr 8 at 2:26

Saamer
2,487 1 3 19

I am not sure how to use it in XAML and with custom renderer? – raV720 Apr 8 at 7:23

@raV720 updated my answer. See the example – Saamer 2 days ago

I know how DI works. It does not solve the problem. My question is about creating control in native platform and use it via custom renderer in xaml. – raV720 2 days ago

@raV720, As it is mentioned here docs.microsoft.com/en-us/xamarin/xamarin-forms/app-fundamentals/… "Every Xamarin.Forms control has an accompanying renderer for each platform that creates an instance of a native control". Are you using Xamarin.Forms controls or libraries, because only XF controls allow you to use Custom Renderers? – Saamer 2 days ago

Please read the question carefully, especially last paragraph. The problem is not so obvious. Of course I use XF controls and native controls as it should be used - as I have mentioned in question it works under some circumstances. – raV720 yesterday