

Forum:[Hack Forums](#)

Thread: [EASYIMGUI | USE IMGUI WITH WINFORMS / .NET FRAMEWORK.](#)

fisica638 - February 3, 2024, 3:15 pm

EasyImGui allows you to Create your GUI with ImGui, without having to configure anything, you don't need any additional knowledge.

You just need to install EasyImGui from NuGet and that's it! .

EasyImGui has two ImGui Views:

ImGuiForm: A Single View, It does not support multi-Windows, and to work, it must be Launched with:

Code `Application.Run(SingleImGuiWindow);` .

MultiImGuiForm: Allows Multiple ImGui Windows, It can be launched normally with Code `.Show()`, It has no restrictions and using WinAPIS, you could embed the windows in Controls Panel or in other Forms.

Note:

If within your ImGui code you launch controls that block the Main UI, such as `OpenFileDialog` or `MessageBox.Show`, it will cause your application to fail, this is easy to solve, Simply do everything Asynchronously, Without Dialogs.

Github : <https://github.com/DestroyerDarkNess/EasyImGui>

Preview

Klonopin - February 4, 2024, 7:58 am

Cool, I don't see the C++ version much harder, but neat to see it working on Csharp. much rather stick with C++ imgui though.

HFDone - February 5, 2024, 9:17 am

saw thread with reply about it c++ look very nice

good job for your c# version

Edit - saw that you used DearImGuiSharp warpper any way well done

Suspect. - February 5, 2024, 2:40 pm

Looks decent, code could be improved by handling your exceptions better but still a good release.

fisica638 - February 7, 2024, 5:04 pm

(Feb 5, 2024 10:17 AM)HFDone Wrote: [\[CITE\]](#) saw thread with reply about it c++ look very nice

good job for your c# version

Edit - saw that you used DearImGuiSharp warpper any way well doneImGui.NET, if you are not using modern C#, properties and methods appear as private/read-only; In languages like VB.NET, all properties also appear as read-only.

On the other hand, DearImGuiSharp, despite some small problems, can be fixed with additional code. The downside would be that it is outdated.

In any case, I'm considering creating a separate branch of the project, which uses ImGui.NET, but only as a test branch.

The idea of EasyImGui is to be as stable as possible with C# and VB.NET, and of course with the .net framework.

Actually, I don't know why I wrote this either, since "[ClickableTransparentOverlay](#)" exists.

EasyImGui would be an alternative to ClickableTransparentOverlay, but much more outdated. since EasyImGui uses Dx9 + .Net Framework + DearImGuiSharp (an obsolete imgui wrapper)

(Feb 5, 2024 03:40 PM)Suspect. Wrote: [\[CITE\]](#) Looks decent, code could be improved by handling your exceptions better but still a good release.

Yes, in fact the bugs I found, I corrected during production. I had to upload several versions to nuget, and each of those versions corrected some bug.

Although the latest version is the most stable. I didn't find any more errors.

about exception handling... I was lazy XD, as was commenting on the code.

Well actually I thought I didn't need it, since any errors when creating the device and during rendering will be handled by SharpDX. and displayed as a message. everything else that could fail would be the hook to implement ImGui.

Klonopin - February 7, 2024, 6:11 pm

(Feb 7, 2024 06:04 PM)fisica638 Wrote: [\[CITE\]](#) ImGui.NET, if you are not using modern C#, properties and methods appear as private/read-only; In languages like VB.NET, all properties also appear as read-only.

On the other hand, DearImGuiSharp, despite some small problems, can be fixed with additional code. The downside would be that it is outdated.

In any case, I'm considering creating a separate branch of the project, which uses ImGui.NET, but only as a test branch.

The idea of EasyImGui is to be as stable as possible with C# and VB.NET, and of course with the .net framework.

Actually, I don't know why I wrote this either, since "[ClickableTransparentOverlay](#)" exists.

EasyImGui would be an alternative to ClickableTransparentOverlay, but much more outdated. since EasyImGui uses Dx9 + .Net Framework + DearImGuiSharp (an obsolete imgui wrapper)

Yes, in fact the bugs I found, I corrected during production. I had to upload several versions to

nuget, and each of those versions corrected some bug.

Although the latest version is the most stable. I didn't find any more errors.

about exception handling... I was lazy XD, as was commenting on the code.

Well actually I thought I didn't need it, since any errors when creating the device and during rendering will be handled by SharpDX. and displayed as a message. everything else that could fail would be the hook to implement ImGui.

Why not just use C++ imgui and directx11 for your application?

fisica638 - February 8, 2024, 6:54 am

(Feb 7, 2024 07:11 PM)Klonopin Wrote: [\[CITE\]](#) Why not just use C++ imgui and directx11 for your application?

Doing this in c++ is very easy, everyone does it in c++. The idea of this Project is to be able to do everything in .NET, with C# or VB.NET, that is the real challenge.

Klonopin - February 8, 2024, 6:57 am

(Feb 8, 2024 07:54 AM)fisica638 Wrote: [\[CITE\]](#) Doing this in c++ is very easy, everyone does it in c++. The idea of this Project is to be able to do everything in .NET, with C# or VB.NET, that is the real challenge.

Hm, sure. if you're just doing it for fun I guess. but there's no point when you've got Winforms & WPF if you're actually trying to build a project.

WinForms and WPF are as customizable as ImGui.

fisica638 - February 8, 2024, 7:07 am

If you see it that way, you're right. But remember that this is an addition, not a replacement to Winform/WPF.

You see this, as an obsession of mine, that I always wanted to use imgui, but it had limitations, since my knowledge only covered VB.NET and .NET Framework. now after learning C# I finally managed to do it.