

Easy GUI

- Installation Guide -

Bogdan-Nicolae David
david.bnicolae@gmail.com

March 4, 2023

Contents

1	Installing a release package	3
1.1	Windows	3
1.2	Linux & MacOS	3
2	Building from source	3
3	IDE Setup	3
3.1	Code::Blocks	3
3.2	Visual Studio	5

1 Installing a release package

All the releases can be found on our Github page (<https://github.com/HV0rtex/EasyGUI>) under the **Releases** section. Each release will have pre-built packages for each OS (Operating System). The changes included in each release can be found in the **Changelog.md** file. Depending on your OS and compiler, you may have to download a different package.

1.1 Windows

On Windows OS, the release package and the compiler version have to match **100%**. Any mismatch between the release package and the compiler could lead to linkage and / or compilation errors. As of right now, the following compilers have a prebuilt release package for 64bits:

- Visual C++ 17
- Visual C++ 16
- MinGW GCC 7.3.0

Before installing the extension itself, make sure you have the right version of **SFML** installed. For more details on installing SFML, please checkout their page (<https://www.sfml-dev.org/index.php>). After downloading the extension and installing SFML, please checkout the **IDE Setup** section in order to set everything up for your IDE.

1.2 Linux & MacOS

On Linux and MacOS, downloading the release package will suffice. Before installing the package, make sure you have **SFML** installed. For more details on installing SFML, please checkout their page (<https://www.sfml-dev.org/index.php>). After downloading the extension and installing SFML, please checkout the **IDE Setup** section in order to set everything up for your IDE.

2 Building from source

If you have to build your extension from source, make sure you have the following dependencies installed:

- C++ Compiler
- CMake 3.20+
- SFML 2.5.1+

After installing the dependencies, go to the **scripts** directory and run the build script. The script will build and install the library.

3 IDE Setup

The steps presented bellow occur after downloading and unzipping the release package / building the project from source.

3.1 Code::Blocks

In order to tell the compiler where to look for the header files, right click on the project and select **Build Options**. Select search directories and type in:

1. Under *Compiler*, the path to the **include** directory. 1
2. Under *Linker*, the path to the **lib** directory. 2

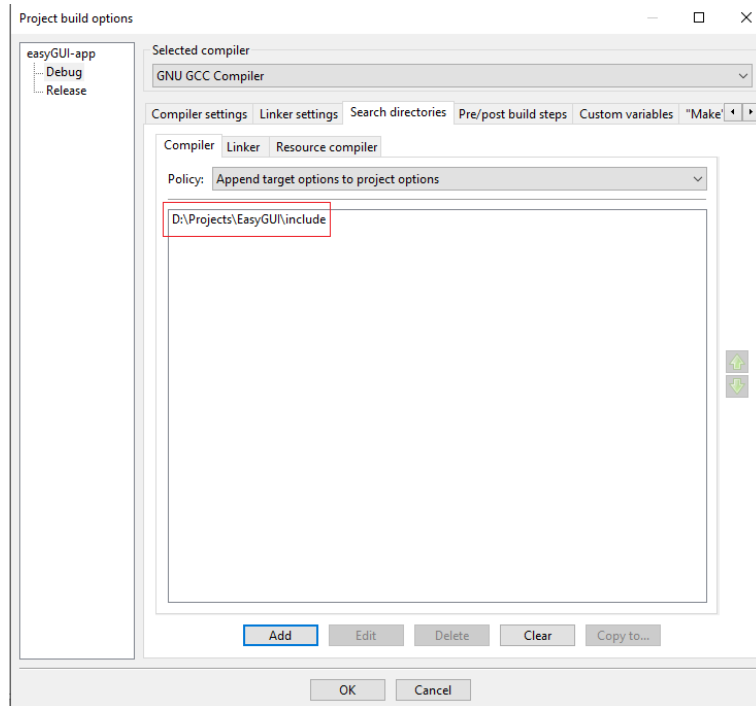


Figure 1: Configuring compiler

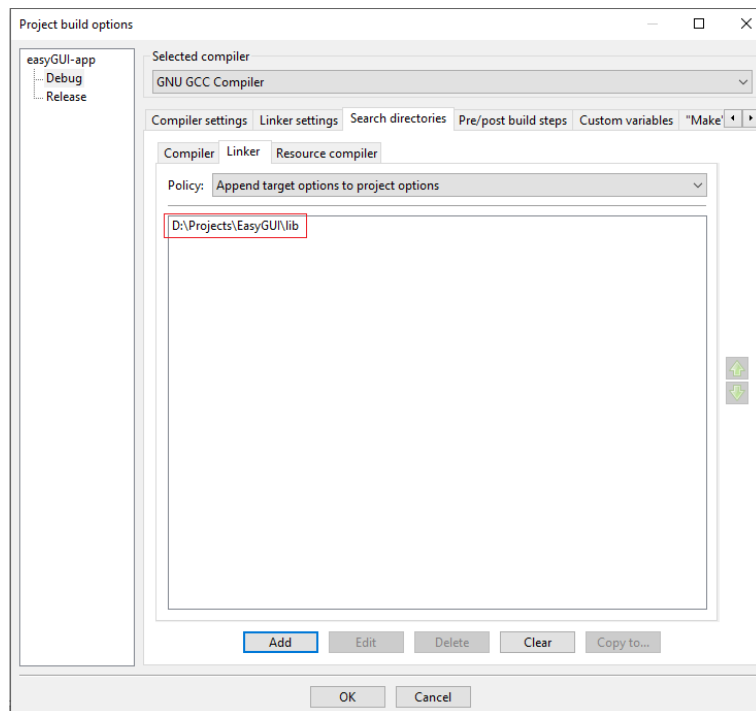


Figure 2: Configuring linker

Afterwards, you need to tell the linker which additional libraries to link to your project. This can be done under **Linker settings** like in the picture below:

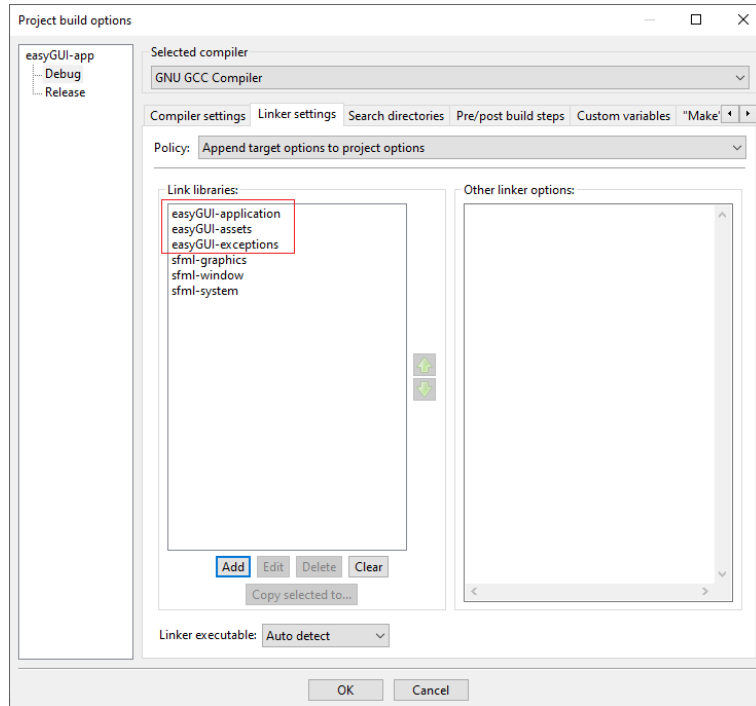


Figure 3: Adding libraries

3.2 Visual Studio

Right click on the solution and select **Properties**.

1. Under *C/C++ - General - Additional Include directories* add the path to the **include** folder.
2. Under *Linker - General - Additional Library directories* add the path to the **lib** folder.

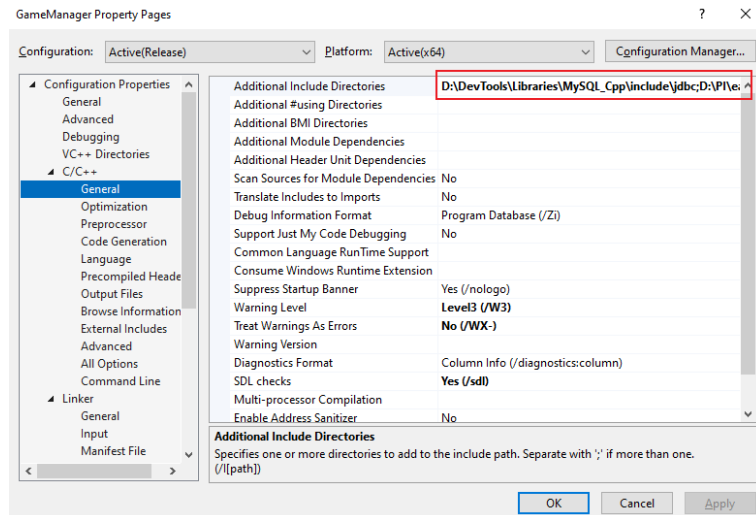


Figure 4: Configuring compiler

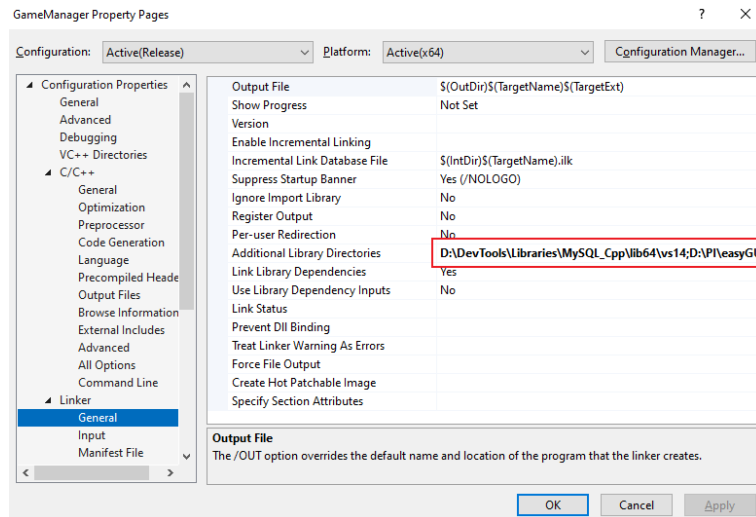


Figure 5: Configuring linker

Afterwards you need to tell the linker which additional libraries to include to your project. This can be done by specifying the libraries under the *Linker - Input - Additional Dependencies* section like so:

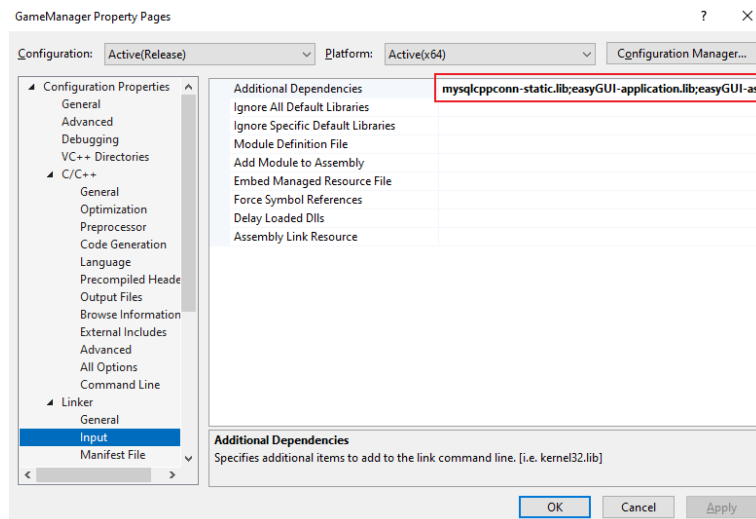


Figure 6: Adding libraries