These instructions are used to recreate a fresh Microsoft Visual Studio project using only the sources available from the git Repository. It has been tested with Visual studio 2022 and CUDA SDK 12.3.

1. Create a new Visual Studio Solution with an **Empty C++ Project** and set its name to "**MapcoreCuda**"
2. Add new Project to the solution – Choose **CUDA 12.3 Runtime** and set its name to "**MapcoreLib**"
3. Delete the default kernel.cu file from MapcoreLib
4. Add new item (Existing items) to the MapcoreLib project and select **all .cu** and **.cuh** files from the MapcoreLib folder of the git repository.
5. Go to properties of the MapcoreLib Project and change **all** of the following items: (Make sure to always change configuration for "**Release**" and not "**Debug**")
    a. General → Change from "**Application (.exe)**" to "**Dynamic Library (.dll)**"
    b. VC++ Directories
        i. **Include Directories**:
            1. Add "C:\Program Files\MATLAB\R2023b\extern\include"
        ii. **Library Directories**:
            1. Add "C:\Program Files\MATLAB\R2023b\extern\lib\win64\microsoft"
    c. CUDA C/C++
        i. Code Generation: Add Support for all GPU architectures, that are required, the recommended list for CUDA SDK 12.3 is below:
        compute_50,sm_50
        compute_52,sm_52
        compute_53,sm_53
        compute_60,sm_60
        compute_61,sm_61
        compute_62,sm_62
        compute_70,sm_70
        compute_72,sm_72
        compute_75,sm_75
        compute_80,sm_80
        compute_86,sm_86
        compute_87,sm_87
        compute_89,sm_89
        compute_90,sm_90
    d. Linker → Input → Additional Dependencies
        i. Add "**cufft.lib**", "**cublas.lib**"
6. Add new item (Existing Item) to the MapcoreCuda project and select the only **.cpp** file from the MapcoreCuda folder of the git repository.
7. Right click the MapcoreCuda project and select "Build dependencies". Include "**CUDA 12.3 (.targets, .props)**" from the available choices.

8. Go to properties of the MapcoreCuda Project and change all of the following items: (Make sure to always change configuration for “**Release**” and not “**Debug**”)
    a. General → Change from “**Application (.exe)**” to “**Dynamic Library (.dll)**
    b. Advanced → Change Target Extension from “**.exe**” to “**.mexw64**”
    c. Linker→ Command line → Additional options
        i. Paste “**/export:mexFunction**”
    d. Linker→Input → Additional Dependencies and add the following items:
        i. “**libmat.lib**”
        ii. “**libmex.lib**”
        iii. “**libmx.lib**”
        iv. “**MapcoreLib.lib**”
    e. VC++ Directories
        i. Include directories:
            1. Add “C:\Program Files\MATLAB\R2023b\extern\include”
            2. Add path to the folder, which contains the MapcoreLib’s **.cuh** and **.cu** files.
        ii. Library Directories
            1. Add “C:\Program Files\MATLAB\R2023b\extern\lib\win64\microsoft”
            2. Add path that points to the release location of the project and that eventually (After building) contains the “MapcoreLib.lib” and “MapcoreLib.dll” files - usually in the form of “ **...**MapcoreCuda\x64\Release ”.
9. Right click on MapcoreCuda project and select “Build Dependencies” → “Project Dependencies”. Check the **MapcoreLib** – so that each time MapcoreCuda is build, it automatically triggers the build of the library itself.
10. If the MapcoreCuda is not set as the **Startup Project** of the Solution, mark it as a startup project of the solution (Right Click and “Set as Startup Project”).
11. **Rebuild** the MapcoreCuda project.
    a. **MapcoreLib.dll** and **MapcoreCuda.mexw64** files should be available in the release folder.