

Moonshot One — On Your Marks!

Getting the stuff you need for Moonshot (for Windows)

Before you can start building web apps in Moonshot, there are a few things to get first. It is best to get them in order:

1. [Visual Studio 2013 Community Edition](#)—all you’re really after is the MS C-compiler.
2. [Eiffel Studio GPL](#)—your language platform and IDE¹.
3. [GitHub Desktop App](#)—so you can get all of the required Moonshot repos².

Install each of the items above in their most basic and default installations—just follow the bouncing balls. Please do them in the precise order so everything “just works.” Don’t worry—if something screws up, we’ll do our best to provide documentation (as we go along) for troublesome setups!

Some Native Tribal Knowledge

Although you are installing Moonshot on Windows, the framework is cross-platform. Soon, you will have full instructions (and capacity) to code Moonshot web sites and apps on Linux and Mac. Hang on—we’re werkin’ on it! Until then, we’re in Windows.

The base language for Moonshot is C through a trans-compiler called Eiffel using Eiffel Studio as the compiler. The basic idea is simple: The Moonshot native code in Eiffel is compiled to C/C++, which is then compiled to binaries, DLLs, and other targets using a C/C++ compiler. That’s where the Microsoft C-compiler (MSC) and Visual Studio come in. You can also use GCC or MinGW, but we recommend using the Microsoft C-compiler because—“it just works better!”

Because of the matters above, this is why we have you fetch Visual Studio first. By doing so, the Eiffel Studio installer “sees” the MSC and uses it right out of the box. Nothing left for you to do! Once you have Visual Studio installed, it will want to open it. Just close it.

Finally—the Eiffel Studio installer will finish. If it wants to open Eiffel Studio, then that is okay. You’ll be doing that a lot soon enough, but for now—open it up, look around, and then close it. We’ll be back to it in a moment, but for now, let’s just get it out of the way.

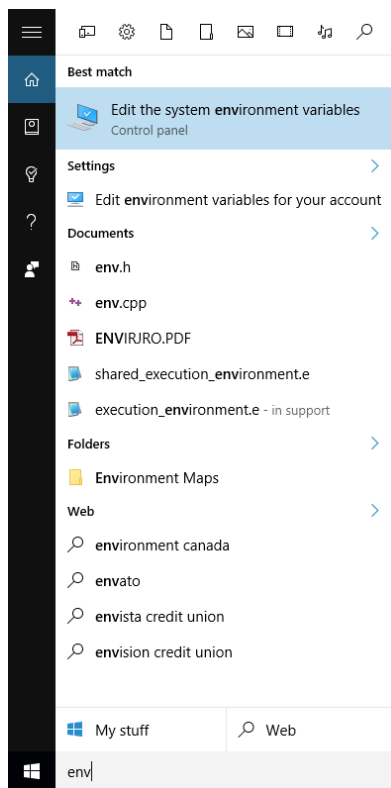
¹ At [Eiffel.org](#) you will see a prominent “Download” button. Don’t do it! Windows will get better with 64-bits soon, but today it is still a hassle (i.e. SYSWOW and so on). Just get the 32-bit version for now (see the link above).

² Coming Soon: We are working on an installer that will bring along everything you need from the repos.

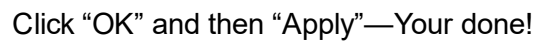
Setting a needed Environment Variable

After installing GitHub Desktop, you will want to create a new Windows environment variable called GITHUB. This is because the Moonshot libraries are all GitHub repo's, so they expect to live in that environment in your host computer. The GitHub Desktop installer automatically creates a GitHub folder in your Documents folder, so there is where you'll point your GITHUB environment variable.

Start by opening the environment variable editing tool from the Windows Control Panel.



This will open the following dialog:



GITHUB is now set and ready to go.

All those Moonshot Repos!

Your last step is to fetch all of the Moonshot code repositories from GitHub. Here, you have a choice on how to get them. Below is a list of all the required repos that you need to clone to your local GITHUB folder. You can walk that list, cloning each one as you need to. The other choice is to run a cmd script that has the same Git-clone commands, but all in one go!

You can find the DOS command prompt script [here](#).

Download the raw file to the root of your \$GITHUB folder in your My Documents folder. Get to a DOS command prompt, change directories to the \$GITHUB folder, and then run the CMD script: [git_moonshot.cmd](#).

If GitHub Desktop has been installed and the computer is able to access the git.exe, then the CMD script ought to clone in all of the Moonshot repositories.