# Computer Graphics and Visualization

Academic year 2025-2026

# Software configuration (MS Windows)

## Introduction

Steps for installing the development environment that we are going to use in the laboratory sessions: the CLion IDE developed by JetBrains.

## CLion installation

> *This video was recorded a while ago, and the look of the JetBrains website has changed a bit, but the steps are similar to those described below:(Spanish)*
>
> *https://youtu.be/bnD8a7uSXYE*

CLion is a cross-platform C/C++ software development IDE (Integrated Development Environment), which can be installed free of charge for educational purposes. You need to create a JetBrains user account through the following link:

https://www.jetbrains.com/shop/eform/students

> **IMPORTANT:** use the university account @red.ujaen.to create the JetBrains account, thus, they can identify you as a student.

Once the user account is created, you can enter your JetBrains profile and link it to your university Google account, if it is more convenient for you, through the following link:

https://account.jetbrains.com/login

Once you have created your JetBrains profile, we recommend downloading the JetBrains Toolbox through this link:

https://www.jetbrains.com/toolbox-app

When you run the Toolbox, you must access the menu (upper right corner) and link it with your JetBrains account (see figures 1 and 2):

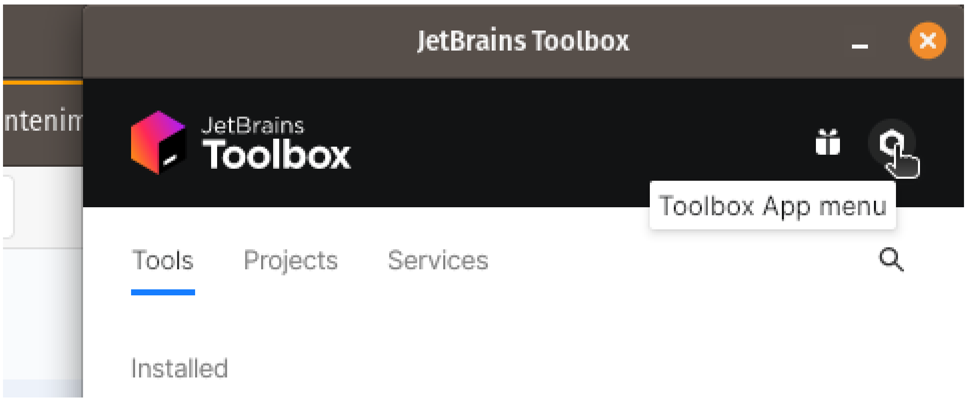


Figure 1

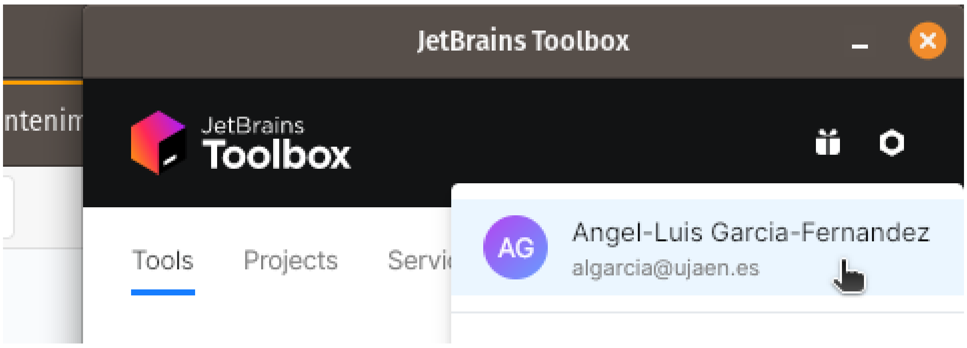


Figure 2

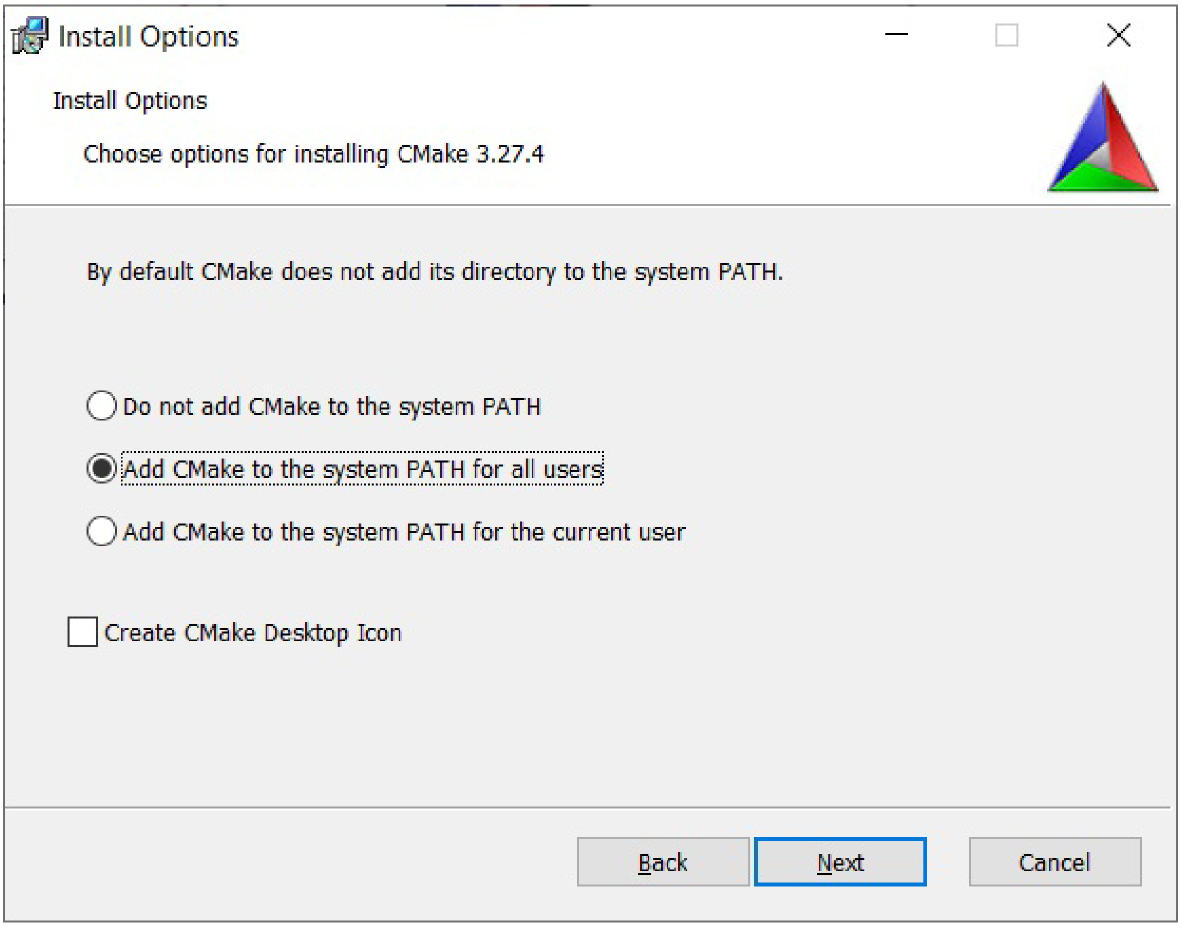
Then, from the Toolbox you can install CLion.

## CMake installation

CMake is a tool for automating software construction processes. It's free, and you can download the installer for Windows from its website:

https://cmake.org/download

After downloading, run the program (it may be necessary to disable the antivirus during the installation) and follow the steps. It is **very important** that during the installation you indicate that CMake is included in the path of the operating system to all users (Figure 3):



Install Options

Install Options

Choose options for installing CMake 3.27.4

By default CMake does not add its directory to the system PATH.

○ Do not add CMake to the system PATH

● Add CMake to the system PATH for all users

○ Add CMake to the system PATH for the current user

□ Create CMake Desktop Icon

Figure 3

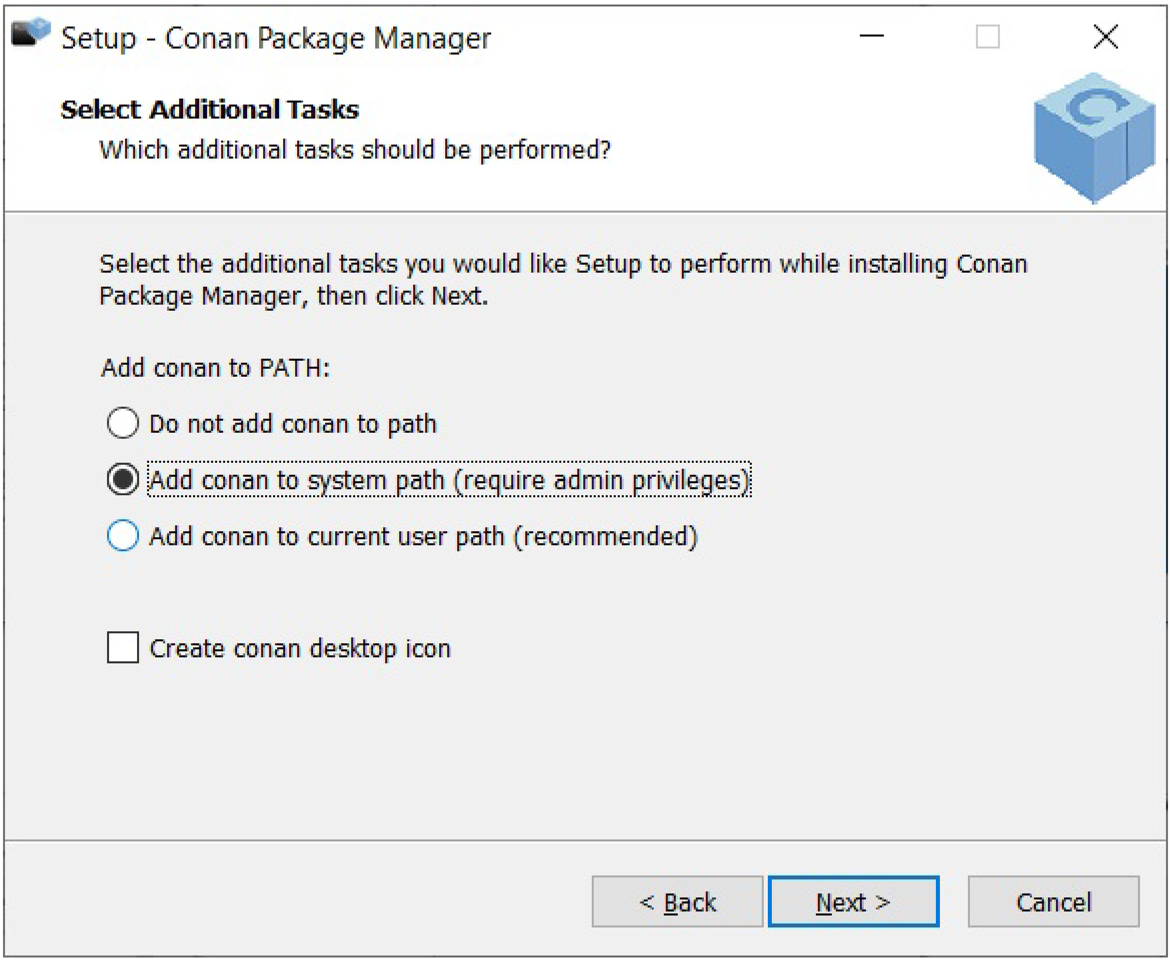
Restart the computer to apply the changes in the path.

## Conan installation

Conan is a package manager for C/C++ that makes it easier for us to install the libraries necessary for compiling our source code. We can download the installer for Windows for free from their website:

https://conan.io/downloads

Once downloaded, we will run it (it may be necessary to disable the antivirus during installation), and as in the case of CMake, it will be necessary to indicate that conan is added to the system path (Figure 4).



Setup - Conan Package Manager

**Select Additional Tasks**

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Conan Package Manager, then click Next.

Add conan to PATH:

○ Do not add conan to path

● Add conan to system path (require admin privileges)

○ Add conan to current user path (recommended)

□ Create conan desktop icon

Figure 4

Restart the computer to apply the configuration changes.

NOTE: In the command shell, run the next command:

*conan profile path default*

and you will get the path of the *default* profile. It should look like as follows:

```
[settings]
arch=x86_64
build_type=Release
compiler=gcc
compiler.cppstd=gnu17
compiler.libcxx=libstdc++11
compiler.version=11
os=Windows
```

If it uses *msvc* instead of *gcc*, edit and change the *default profile*.

# Conan plugin for CLion

In the CLion configuration it will be necessary to install a plugin so that it communicates with the conan installation that we have just made. We can access the CLion settings through the application menus (Figure 5), and search for the corresponding plugin in the marketplace (Figure 6)

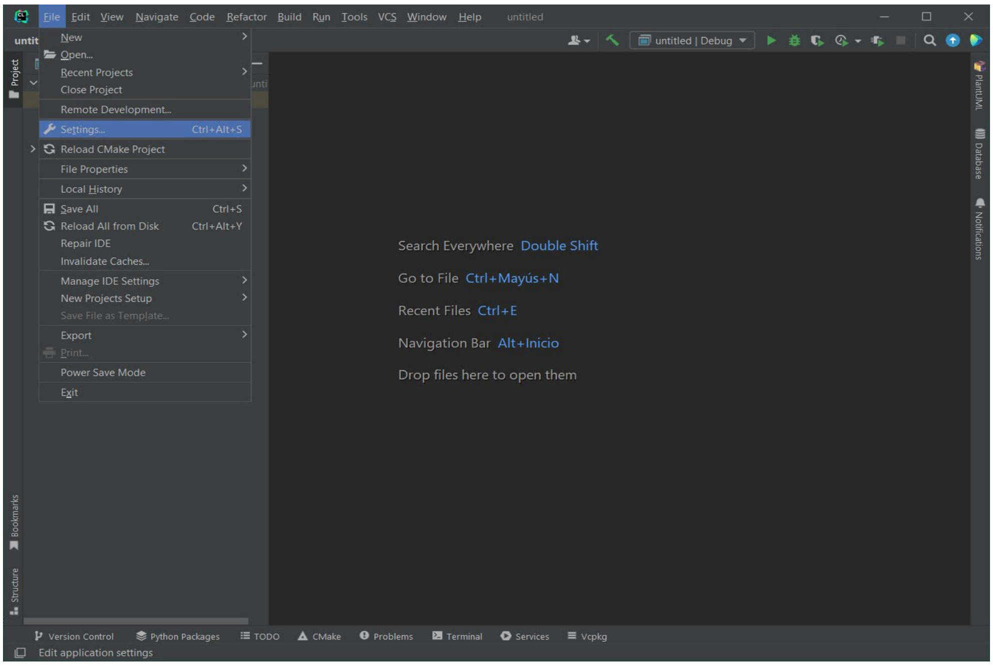
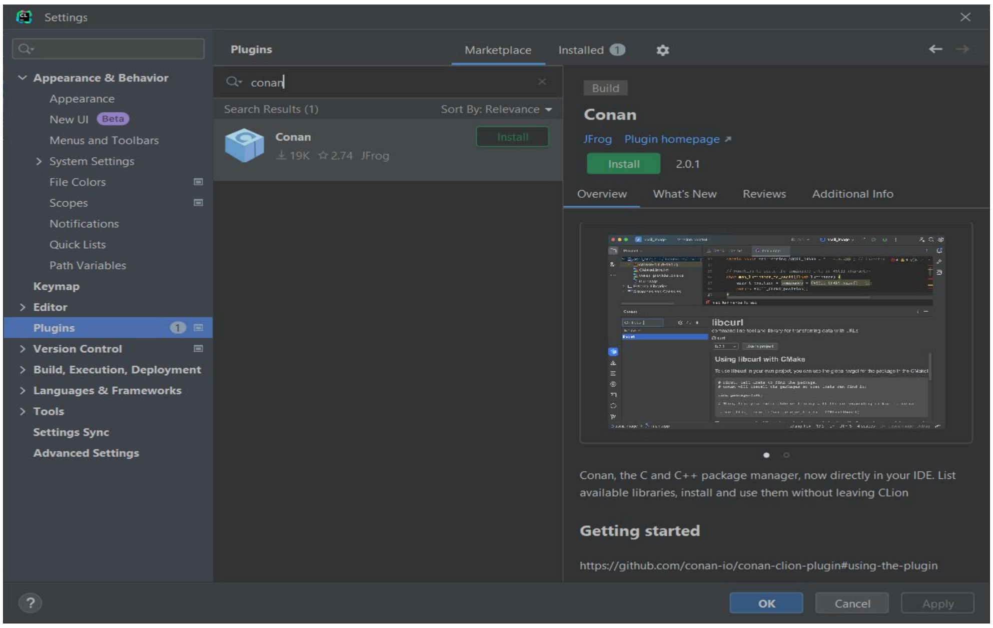


Figure 5



Conan, the C and C++ package manager, now directly in your IDE. List available libraries, install and use them without leaving CLion

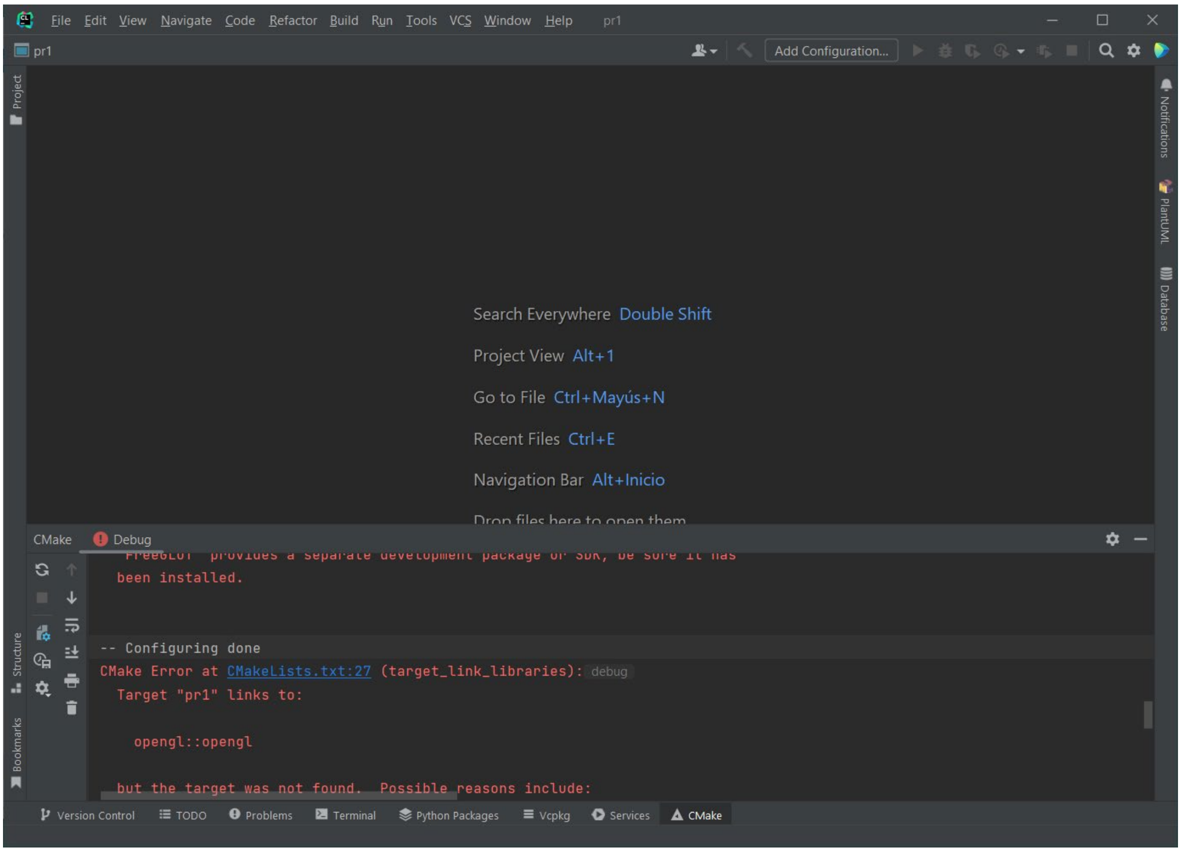
Getting started

https://github.com/conan-io/conan-clion-plugin#using-the-plugin

Figure 6

# Test projects in the new environment

The projects of the practices are already configured with the corresponding libraries. When you open them with CLion, you will see that some CMake errors appear (Figure 7) which are due to the lack of configuring the conan plugin for the project. To do this, you just have to access the conan tab in the bottom panel of CLion, click on the settings button (the wrench - Figure 8) and check the “Use system conan installation” option (Figure 9). Once this is done, the project will be automatically reconfigured, and you will be able to run and work with your projects.



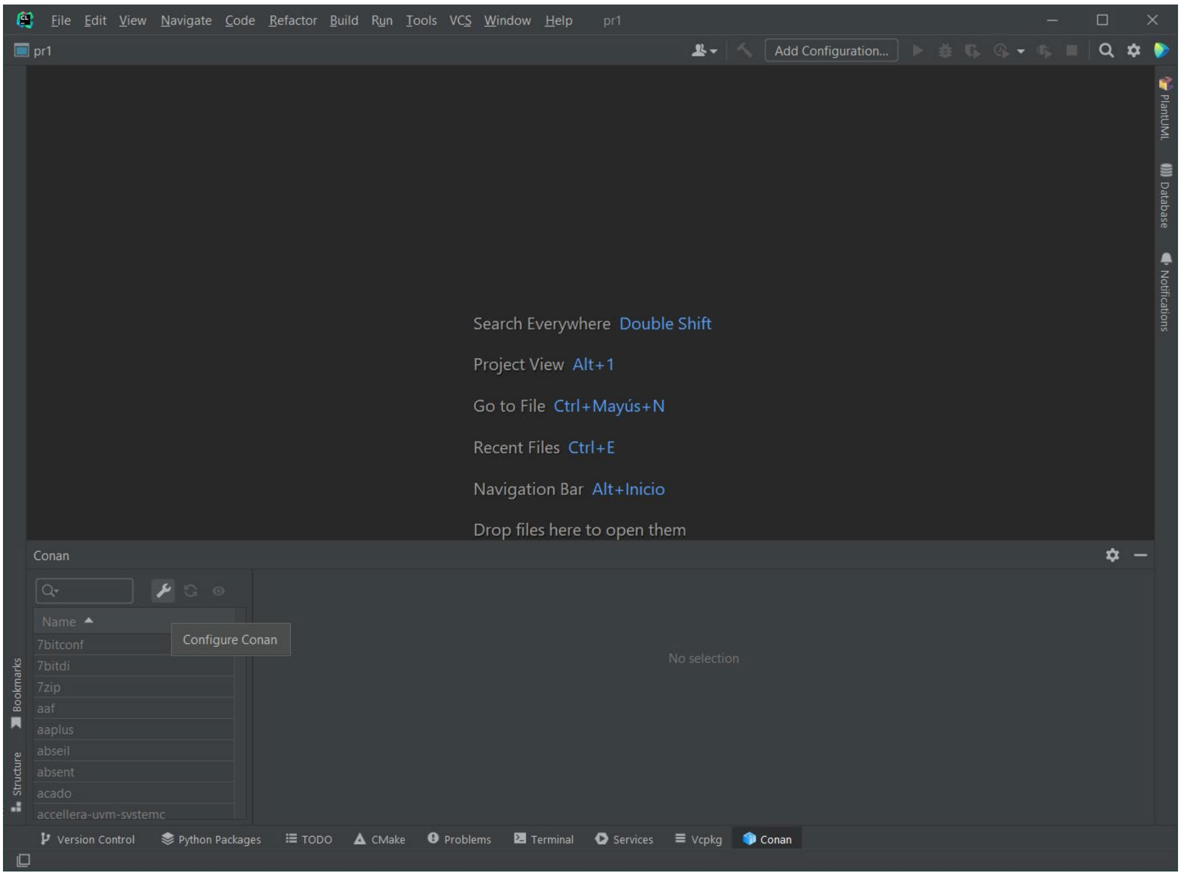
```
   FreeGLUT provides a separate development package or SDK, be sure it has
  been installed.


-- Configuring done
CMake Error at CMakeLists.txt:27 (target_link_libraries): debug
  Target "pr1" links to:

    opengl::opengl

  but the target was not found.  Possible reasons include:
```

Figure 7



Search Everywhere Double Shift

Project View Alt+1

Go to File Ctrl+Mayús+N

Recent Files Ctrl+E

Navigation Bar Alt+Inicio

Drop files here to open them

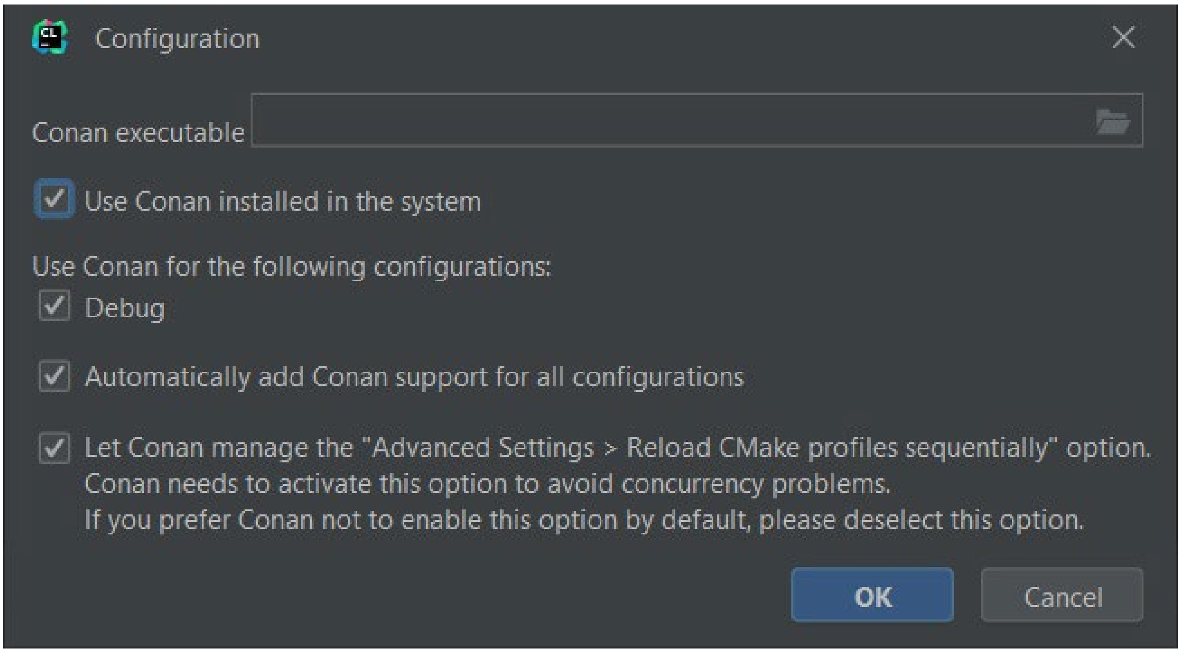
Conan

Configure Conan

| Name ▲ |
|---|
| 7bitconf |
| 7bitdi |
| 7zip |
| aaf |
| aaplus |
| abseil |
| absent |
| acado |
| accellera-uvm-systemc |

No selection

Figure 8



Configuration

Conan executable

☑ Use Conan installed in the system

Use Conan for the following configurations:

☑ Debug

☑ Automatically add Conan support for all configurations

☑ Let Conan manage the "Advanced Settings > Reload CMake profiles sequentially" option.
Conan needs to activate this option to avoid concurrency problems.
If you prefer Conan not to enable this option by default, please deselect this option.

Figure 9

## Creating new projects

The use of the conan plugin is described on this website. Here's how to add libraries to a project automatically:

https://blog.conan.io/introducing-new-conan-clion-plugin

En nuestros proyectos necesitaremos instalar las bibliotecas

- OpenGL
- OpenGL registry
- FreeGLUT

Tal y como se explica en la web, solo tienes que:

- Cambiar la configuración del plugin para indicarle que utilice la instalación de conan del sistema (Figuras 8 y 9).
- Buscar en la ventana del plugin de conan las bibliotecas.
- Añadirlas al proyecto haciendo clic en el botón de “Usar en el proyecto”
- Será necesario hacer cambios en el archivo CMakeLists.txt del proyecto. En los proyectos de partida que te damos puedes ver ejemplos de configuración que puedes adaptar.

In our projects we will need to install the libraries

- OpenGL
- OpenGL registry
- FreeGLUT

As explained on the website, you just have to:

- Change the plugin settings to use the system conan installation (Figures 8 and 9).
- Search the conan plugin window for libraries.
- Add them to the project by clicking the “Use in project” button
- Changes will need to be made to the project's CMakeLists.txt file. In the starting projects, you can see configuration examples that you can adapt.

Universidad de Jaén
Departamento de Informática

# Software installation/configuration for GNU/Linux systems based on Debian/Ubuntu

## Introduction

On these systems, there is no need to use conan to install the necessary libraries.

## Libraries installation

To install the libraries, it will be enough to use the package manager. The quickest way is through a terminal.

If we do not yet have the application development tools installed (compiler, debugger...), we can install them by running: *sudo apt-get install build-essential*

If we already have the tools installed, by installing the FreeGLUT package and accepting the dependencies it needs, everything necessary is already installed:

```
sudo apt-get install freeglut3-dev
```

## CLion installation

To run the ToolBox, the library *libfuse* should be installed:

```
sudo apt-get install libfuse2
```

The steps for installing CLion are similar to those indicated for Windows. When the ToolBox,is downloaded, it will be in the form of a compressed file in .tar.gz format. Then, unzip and run it.

## Test projects in the new environment

Once the project is opened with CLion, CMake will automatically configure the compilation of the project. If the libraries are installed correctly, they will be added automatically.

## Creating new projects

When the project is created, edit the CMakeLists.txt file to add the libraries that need to be linked. In the original projects of the practices, there are examples that you can transfer to your project.