ATAD | Development Environment

> **Note**: This document assumes you have obtained the required software, either through manual installation or *docker container*.

Throughout the course you'll need to `git clone` *template* repositories. These contain, at minimum, a project structure suitable for developing C programs within VS Code; others, some additional initial code.

This document describes the necessary steps - that you must be confortable with - to clone a repository, code, compile and execute a program:

1. Clone a repository and open the project in VS Code;
2. Install VS Code extensions (just on first run);
3. Code, compile and run the program.

> **Youtube Tutorial**: https://www.youtube.com/watch?v=THsizwp30r0
>
> The described steps of this document are depicted in this video.

1 | Clone a repository and open the project in VS Code

Before starting, we recommend creating a top-folder in your filesystem to centralize all future program folders, e.g., `ATAD_Projects`.

Next, follow the instructions for your software installation method.

1.1 | Windows/WSL or Linux

1. In Explorer navigate to the created top-folder contents and perform *Shift* + *Right-Click* inside. You should see in the context menu an option similar to **Open Linux Shell here**; choose that option and the Ubuntu terminal will open on that folder.
    - If you're using Linux, use the equivalent **Open Terminal** context action to get a terminal.
2. From the terminal, clone the, e.g., `CProgram_Template` repository.

```
$> git clone https://github.com/estsetubal-
 ↪  atad/CProgram_Template.git
 ↪  MyCProgram
```

- Change the URL, according to the required template;
- The last argument (optional) is the name of the folder that will be created with the repository contents. If you do not supply this, the created folder will have the same name as the repository, e.g., "CProgram_Template".

3. From the same terminal, open the project with *VS Code*:

```
$> code MyCProgram/
```

- **Note:** when asked if you would like to "reopen folder to develop in a container", **choose "Don't Show Again"**.

1.2 | Docker container

If you're using the *docker container* development methodology, we'll assume you're on Windows or MacOS.

In either case, you should have installed the `git` application, as described in the Software.pdf document.

1. To clone a repository, follow steps 1-3 of the previous section, but from a PowerShell terminal or MacOS Terminal.
2. When asked if you would like to "reopen folder to develop in a container", **choose "Reopen in container"**.

2 | Install VS Code extensions

Install the following extensions from VS Code Marketplace (use the extension's **ID** to search for it):

- Name: **C/C++**
  ID: ms-vscode.cpptools

Description: C/C++ IntelliSense, debugging, and code browsing.
Publisher: Microsoft

- Name: **Doxygen Documentation Generator**
  ID: cschlosser.doxdocgen
  Description: Let me generate Doxygen documentation from your source code for you.
  Publisher: Christoph Schlosser

> This procedure will install these extensions alongside your WSL or *docker* environment. There is no need to repeat this step afterwards. They will even get updated automatically, when needed.

3 | Code, compile and run the program

Use the IDE to create new files (e.g., modules) and code.

To compile and run your program, open the *integrated terminal*: Menu **Terminal > New Terminal**.

Invoke the makefile:

```
$> make
```

and run the program:

```
$> ./prog
```

Appendix - Tips and tricks

Repository URLs

The repository URL needed for the *clone* procedure can be obtained from the repository page in GitHub, as seen in the following image:

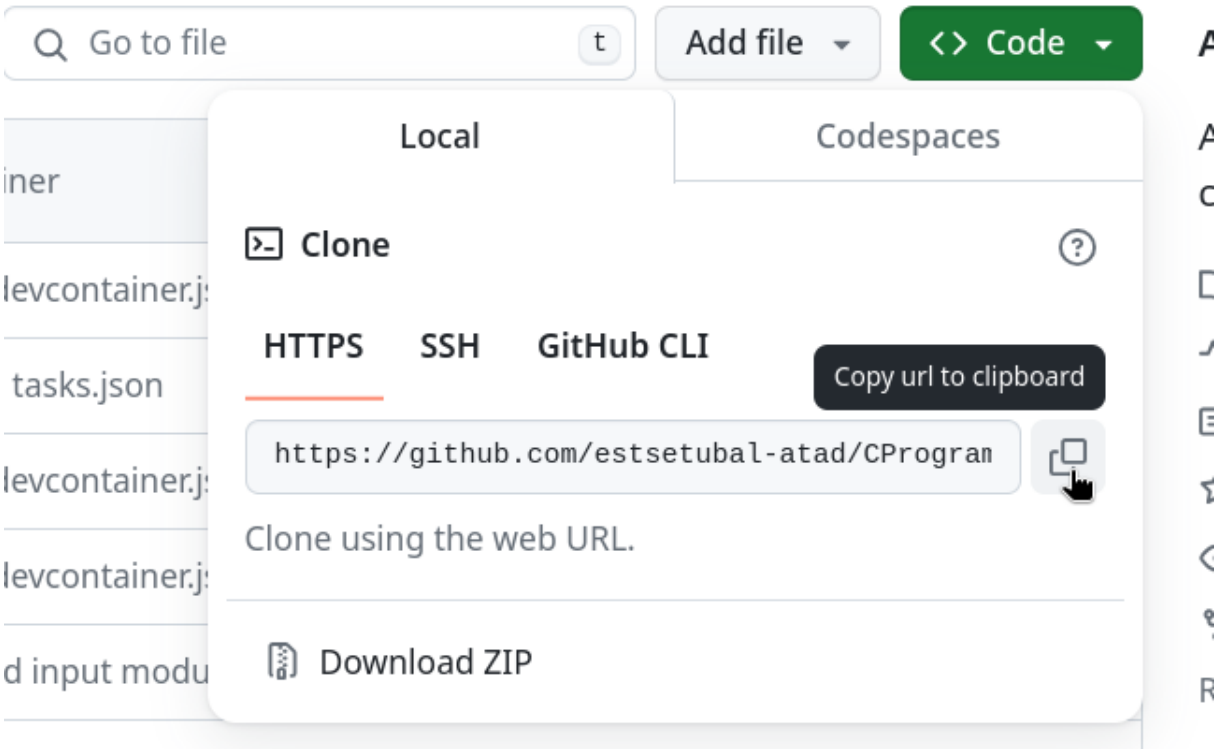

Figure 1: image

Clicking the button will *copy* the URL to your clipboard. You can then *paste* this URL with Control+V, or Control+Shift+V in a Linux/WSL terminal.

Remote connections

Always check that you are opening your projects with a remote connection, either to WSL or a container, on the green bottom left corner of VS Code.