

MANUAL TÉCNICO

JANER ALFONSO PEREJA GUTIERREZ

Ze Roberto Castro Castro

PROGRAMACION III – 5L

UNIVERSIDAD DE INVESTIGACION Y DESARROLLO – UDI

2024

Instalación de Docker Desktop en WSL2

Descarga: Docker Desktop para Windows.

- Habilita la característica de Windows: Hyper-V
- Activa la integración con WSL2 en Docker Desktop.

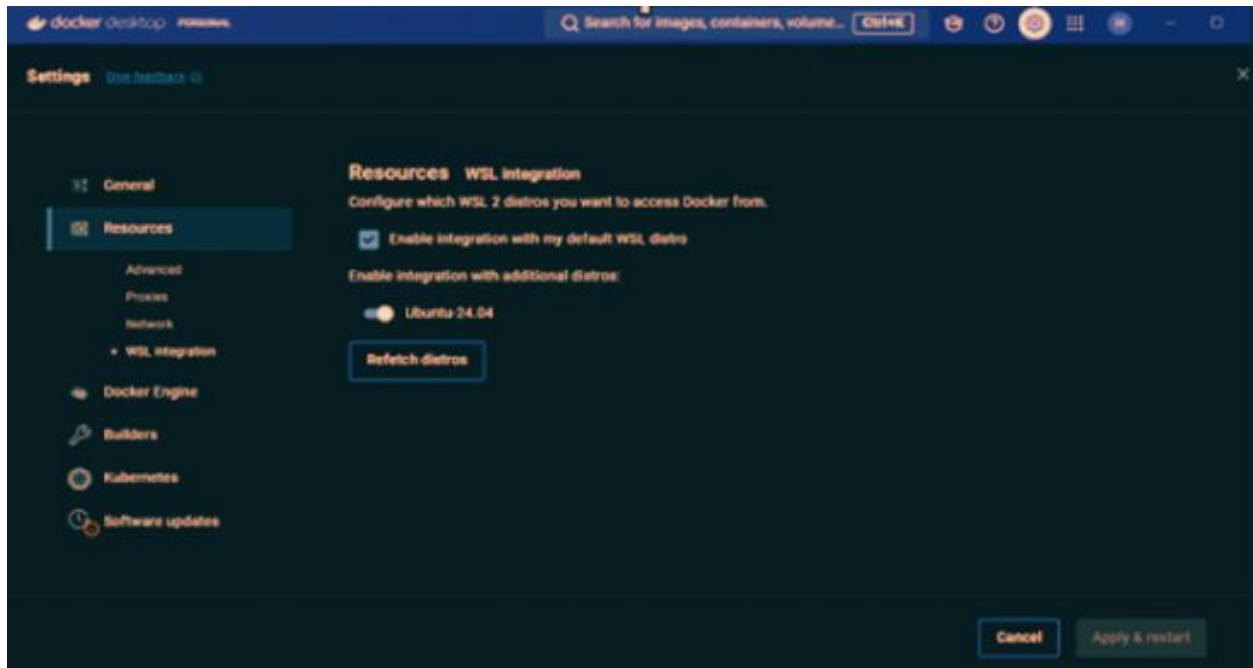
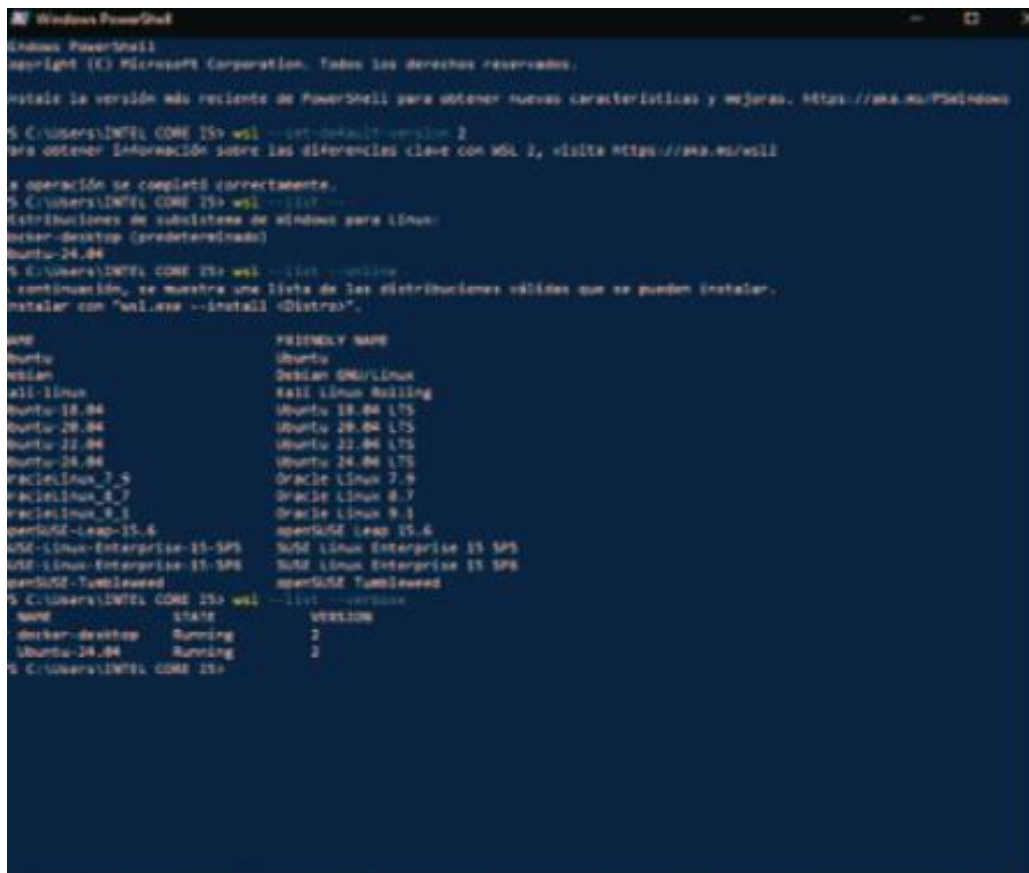


figura 1. Docker Desktop en WSL2

INSTALACION DE WSL2

- `wsl --install`
- `wsl --set-default-version 2`
- Abre una terminal y ejecuta el comando `wsl --list --verbose` para verificar que tu distribución esté ejecutándose en WSL2.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instala la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PowerShell

PS C:\Users\INTEL CORE I5> wsl --set-default-version 2
Para obtener información sobre las diferencias clave con WSL 1, visita https://aka.ms/wsl1

La operación se completó correctamente.
PS C:\Users\INTEL CORE I5> wsl --list --verbose
Distribuciones de subistema de windows para linux:
docker-desktop (predeterminado)
Ubuntu-24.04
PS C:\Users\INTEL CORE I5> wsl --list --online
Continuación, se muestra una lista de las distribuciones válidas que se pueden instalar.
Instalar con "wsl.exe --install <Distro>".

NAME                                FRIENDLY NAME
Ubuntu                              Ubuntu
Debian                              Debian GNU/Linux
kali-linux                          kali linux Rolling
Ubuntu-18.04                        Ubuntu 18.04 LTS
Ubuntu-20.04                        Ubuntu 20.04 LTS
Ubuntu-22.04                        Ubuntu 22.04 LTS
Ubuntu-24.04                        Ubuntu 24.04 LTS
OracleLinux_7.5                     Oracle Linux 7.5
OracleLinux_8.7                     Oracle Linux 8.7
OracleLinux_9.1                     Oracle Linux 9.1
openSUSE-Leap-15.6                  openSUSE Leap 15.6
SUSE-Linux-Enterprise-15-SP5        SUSE Linux Enterprise 15 SP5
SUSE-Linux-Enterprise-15-SP6        SUSE Linux Enterprise 15 SP6
openSUSE-Tumbleweed                 openSUSE Tumbleweed
PS C:\Users\INTEL CORE I5> wsl --list --verbose
NAME                STATE              VERSION
docker-desktop      Running             2
Ubuntu-24.04        Running             2
PS C:\Users\INTEL CORE I5>
```

figura 2. INSTALACION DE WSL2

INSTALACION LARAVEL

Instalación de Composer:

- sudo apt update
- sudo apt install curl php-cli php-mbstring unzip
- curl -sS <https://getcomposer.org/installer> | php
- sudo mv composer.phar /usr/local/bin/composer

Nuevo proyecto:composer create-project --prefer-dist laravel/laravel
projectsLaravel

```
harsa@project:~$ ls
projectsLaravel  projectsLust: 'projectsLust  proje.nse'
harsa@project:~$ cd projectsLaravel
harsa@project:~/projectsLaravel$ ls
projectsLaravel
harsa@project:~/projectsLaravel$ cd projectsLaravel
harsa@project:~/projectsLaravel/projectsLaravel$ ls
README.md  artisan  composer.json  config  docker-compose.yml  phpunit.xml  resources  storage  vendor
app  bootstrap  composer.lock  database  package.json  public  routes  tests  vite.config.js
harsa@project:~/projectsLaravel/projectsLaravel$
```

figura 3. **INSTALACION LARAVEL**

```
zharsa@project:~$ cd projectsLust
zharsa@project:~/projectsLust$ ls
project-rocket
zharsa@project:~/projectsLust$ project-rocket
project-rocket: command not found
zharsa@project:~/projectsLust$ cd project-rocket
zharsa@project:~/projectsLust/project-rocket$ ls
Cargo.lock  Cargo.toml  Dockerfile  Rocket.toml  docker-compose.yml  src  target
zharsa@project:~/projectsLust/project-rocket$ code .
Updating VS Code Server to version 38c31bc77e0dd6ae88a4e9cc93428cc27a56ba40
Removing previous installation...
Installing VS Code Server for Linux x64 (38c31bc77e0dd6ae88a4e9cc93428cc27a56ba40)
Downloading: 100%
Unpacking: 100%
Unpacked 1761 files and folders to /home/zharsa/.vscode-server/bin/38c31bc77e0dd6ae88a4e9cc93428cc27a56ba40.
Looking for compatibility check script at /home/zharsa/.vscode-server/bin/38c31bc77e0dd6ae88a4e9cc93428cc27a56ba40/bin/helpers/check-requirements.sh
Running compatibility check script
Compatibility check successful (0)
zharsa@project:~/projectsLust/project-rocket$
```

figura 4 **ABRIR EL PROYECTO EN VS CODE**

PROYECTO EN VS CODE

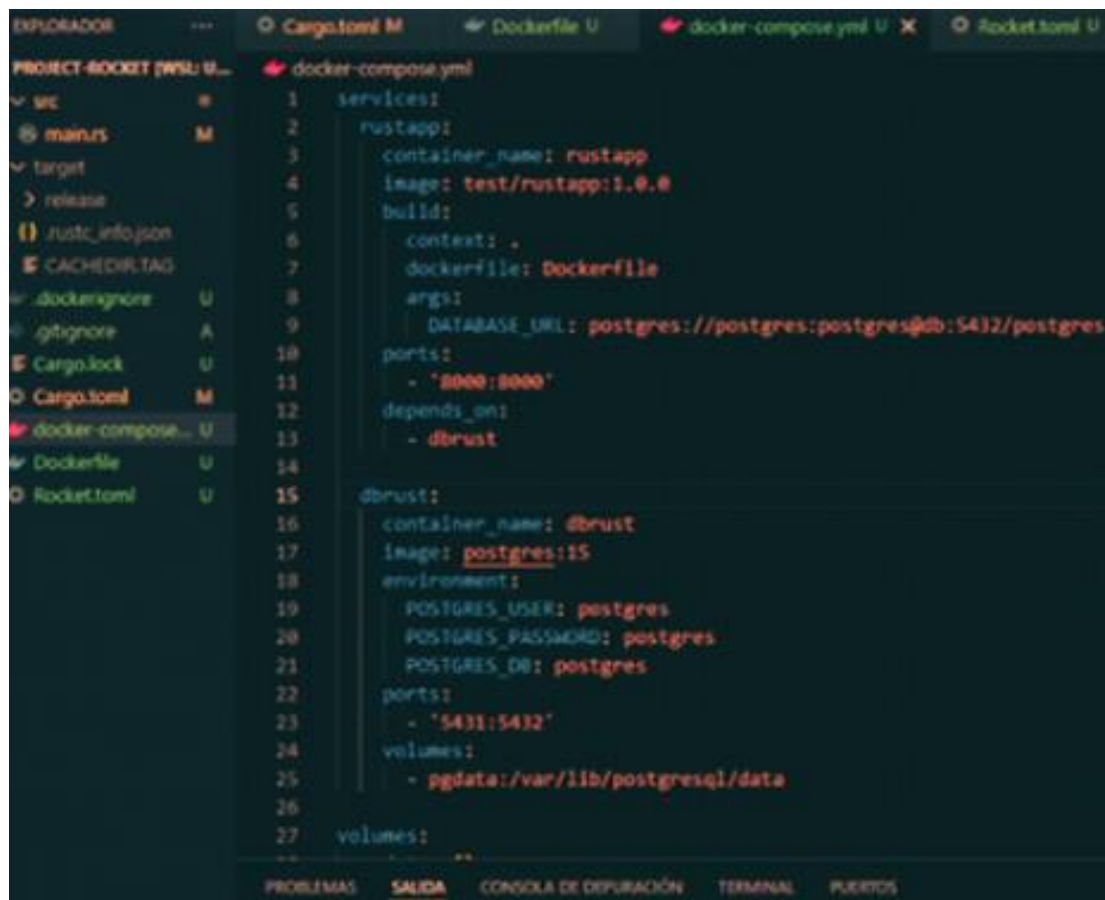


Figura 5. **PROYECTO EN VS CODE**

INSTALACION DEL RUST

ejecuta el siguiente comando.

- `curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh`

verifica la versión de Rust con:

`rustc --version`

`[dependencies]`

`rocket = "0.5.0"`

```
homsag@homsag:~$ rustc --version
rustc 1.80.1 (3f5fd8dd4 2024-08-06)
homsag@homsag:~$ cargo --version
cargo 1.80.1 (376290515 2024-07-16)
homsag@homsag:~$ rustup default stable
info: using existing install for 'stable-x86_64-unknown-linux-gnu'
info: default toolchain set to 'stable-x86_64-unknown-linux-gnu'

stable-x86_64-unknown-linux-gnu unchanged - rustc 1.80.1 (3f5fd8dd4 2024-08-06)
homsag@homsag:~$ cargo new hello-rocket --bin
   Creating binary (application) 'hello-rocket' package
note: see more `Cargo.toml` keys and their definitions at https://doc.rust-lang.org/cargo/reference/manifest.html
homsag@homsag:~$ cd hello-rocket
homsag@homsag:~/hello-rocket$ ls
Cargo.toml  src
homsag@homsag:~/hello-rocket$ code .
```

figura 6. **INSTALACION DEL RUST**

CONTENEDORES EN DOCKER

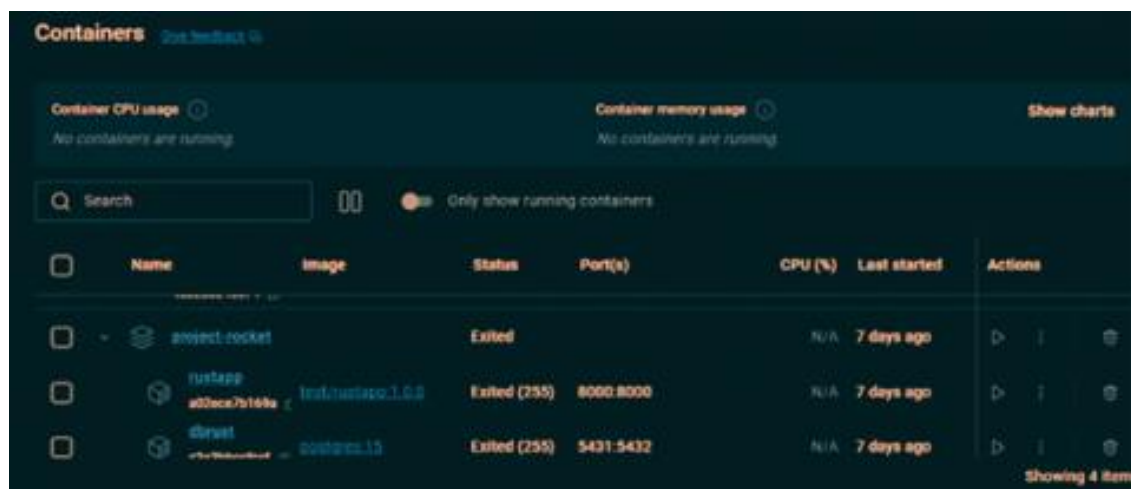


figura 7. **CONTENEDORES EN DOCKER**