**UNIVERSIDAD AUTÓNOMA DE SAN LUIS POTOSÍ**
**FACULTAD DE INGENIERÍA**
**ÁREA DE CIENCIAS DE LA COMPUTACIÓN**
**INGENIERÍA EN SISTEMAS INTELIGENTES**
**PROYECTOS COMPUTACIONALES III**

PROYECTO

# PSICOFI

## MANUAL DE ADMINISTRADOR

> BAUTISTA GÓMEZ JUAN PABLO
> HERNÁNDEZ ALONSO JESÚS ALEJANDRO
> RAMÍREZ MARTÍNEZ FERNANDO ANTONIO
> RAMÍREZ PADRÓN ERICK ENRIQUE
> VEGA MARQUEZ ANDREA ISABEL

ASESORA

**SANDRA EDITH NAVA MUÑOZ**

**DOCENTE**

**EDGAR FRANSISCO CASTILLO BARRERA**

30 DE NOVIEMBRE 2024

# Índice

# Introducción

El propósito de este manual es explicar el proceso de instalación de las herramientas necesarias para que la plataforma PSICOFI funcione correctamente.

PSICOFI es una plataforma web que implementa un sistema integral de gestión de pacientes para el Departamento de Atención Psicológica de la Facultad de Ingeniería de la Universidad Autónoma de San Luis Potosí.

El proyecto está basado en el modelo cliente-servidor, en donde un cliente o usuario solicita información al servidor, el cual en base a la petición del cliente identifica que información se solicita y responde a la petición con la información solicitada en un formato adecuado para que el cliente pueda procesarla y mostrarla.

Se necesitan de 4 herramientas principales para el funcionamiento de la plataforma. **Angular (Cliente)** que se encarga de mostrar visualmente la página, así como el diseño de esta. **Laravel (Servidor)** que recibe las peticiones de la página web para extraer información de la **base** de datos. **MySQL** que contiene la base de datos de la plataforma y por último **GIT** que permite la conexión con el repositorio del proyecto, permitiendo descargar todos los archivos de la página web y del servidor.

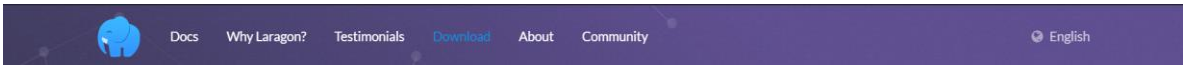
Adicionalmente, al tratarse de una plataforma web, el uso y disponibilidad de internet es indispensable para el correcto funcionamiento de la plataforma.

# Instalación de herramientas

Como se mencionó en la introducción, PSICOFI necesita de 4 herramientas principales (Angular, Laravel, MySQL y GIT). Cada uno de estos, son necesarios para que la plataforma funcione, sin embargo, cada herramienta necesita de instalaciones adicionales para poder funcionar correctamente. Cada una de las herramientas necesarias se listan a continuación, con su respectivo proceso de instalación:

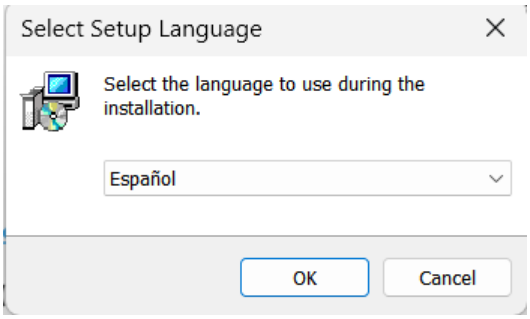
- Laravel
    - Laragon

Laragon es una herramienta que permite la creación de un servidor local para entornos de desarrollo web, para este proyecto el uso de Laragon se destina para el uso de Laravel. Para instalar esta herramienta debemos dirigirnos a la página oficial de Laragon "https://laragon.org/" , para posteriormente entrar en la sección **Download**.

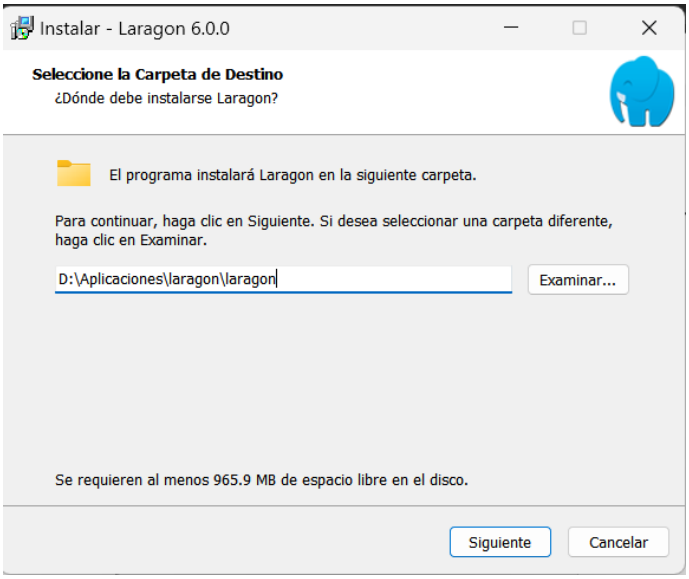


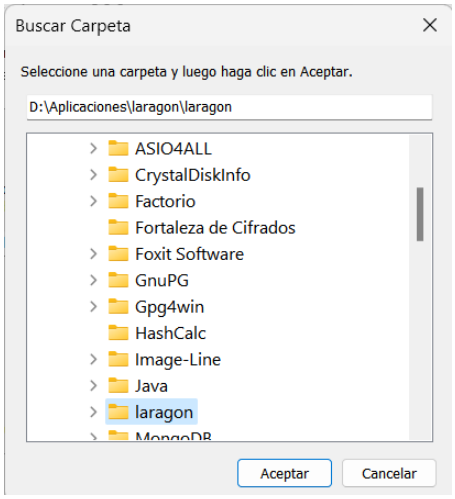

En esta página encontraremos dos versiones **Laragon full** y **Laragon portable**, la que descargaremos es la versión **Laragon full** que contiene PHP y MySQL.


Una vez descargado ejecutaremos el archivo instalación con extensión .exe. Nos mostrará la siguiente ventana, en donde seleccionaremos el idioma con el que se instalará el programa y confirmamos dando clic en el botón **Ok.**

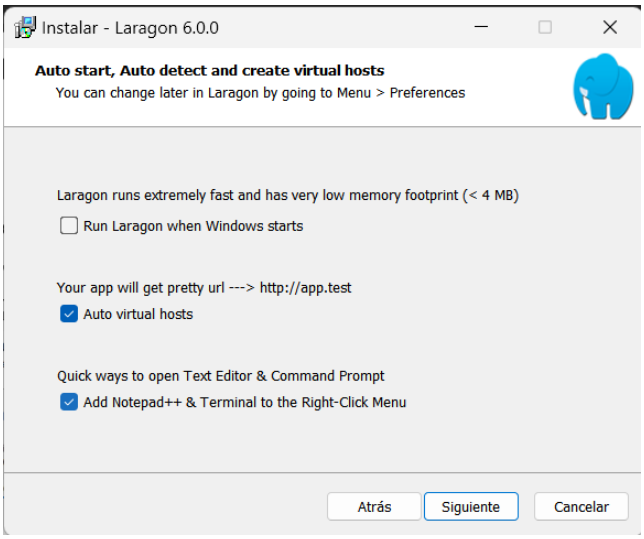



A continuación, seleccionaremos la ruta donde se realizará la instalación. Se recomienda dejar la ruta por defecto, pero si desea cambiarla puede dar clic en el botón **Examinar...** y navegar por el explorador de archivos hasta encontrar la carpeta donde desea realizar la instalación.




La siguiente ventana corresponde a la configuración de algunas funcionalidades, las cuales no es necesario modificar y daremos clic en siguiente.




Finalmente confirmamos la instalación dando clic en instalar y esperamos a que termine la instalación. Al terminar se mostrará una nueva ventana indicando que la instalación fue correcta y damos clic en **Finalizar.**

Si después de unos minutos no se ejecutó automáticamente el programa iniciarlo manualmente, donde nos encontraremos la interfaz principal.

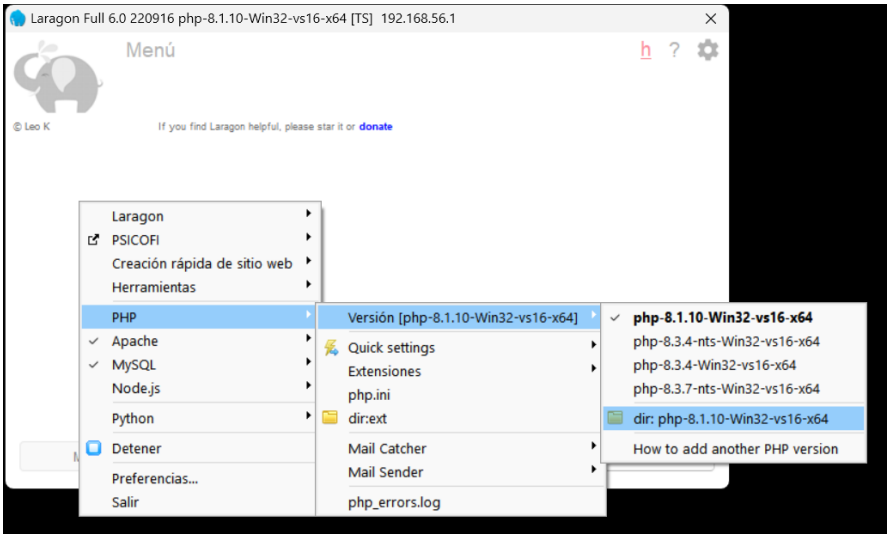

- PHP

Laravel funciona con la versión de PHP 8.1 en adelante, y la versión que se instala con Laragon es la versión 8.0, por lo que necesitaremos actualizarla. PSICOFI se desarrolló con la versión 8.3.4 NTS, sin embargo, PHP se actualiza

constantemente y hasta la fecha de elaboración de este manual, PHP se encuentra en su versión 8.3.7, la cual sigue siendo compatible con el sistema.

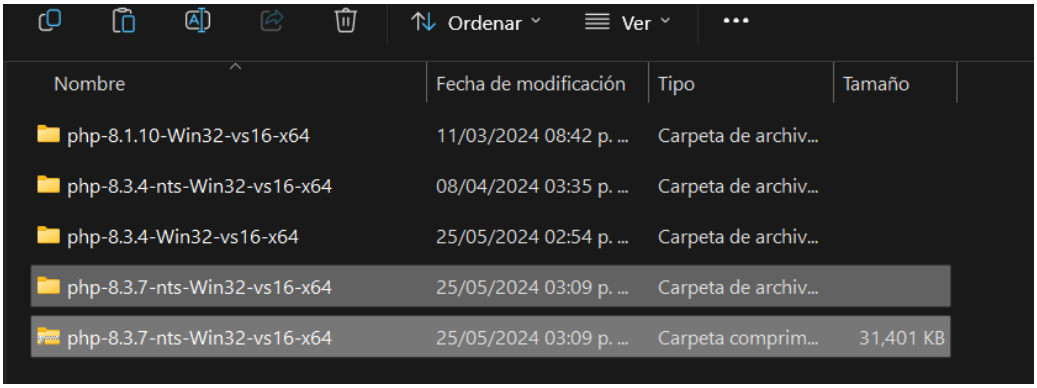
Para actualizar primero tendremos que descargar la nueva versión de PHP (8.3.7) del siguiente enlace "https://windows.php.net/download#php-8.3" , en la página encontraremos dos versiones **Thread Safe** y **Non Thread Safe**. Descargaremos el archivo .zip de la versión **Non Thread Safe.**


En Laragon tendremos que hacer clic en cualquier parte de la interfaz, seleccionar PHP, y a continuación seleccionar la opción que comienza con **dir.**

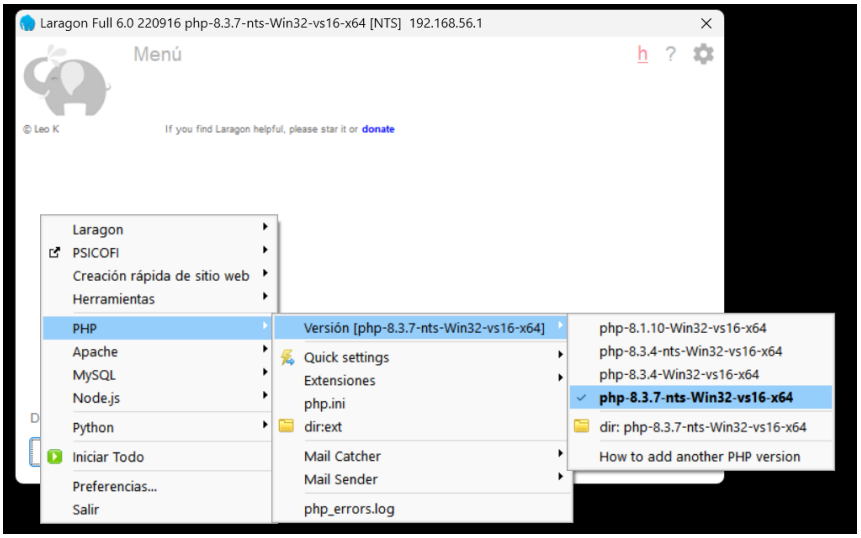



Automáticamente se abrirá el explorador de archivos de Windows, en la dirección donde se guardan las versiones de PHP en Laragon, en esa carpeta

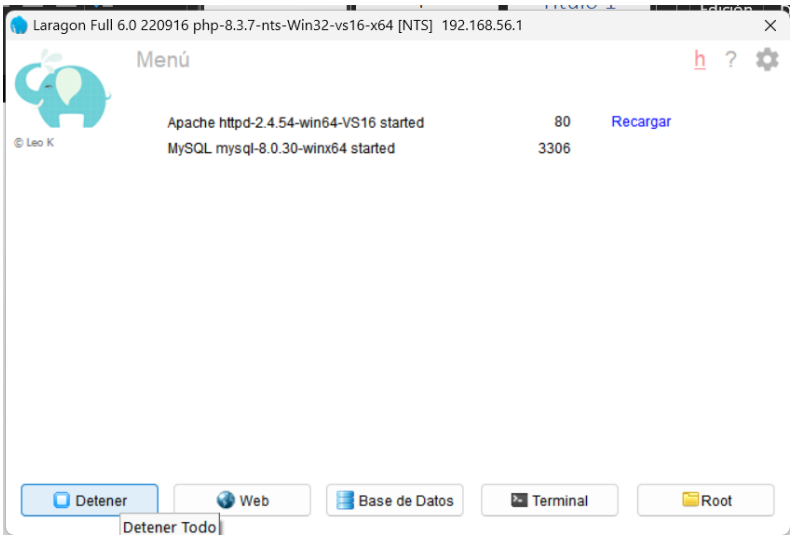
colocaremos el archivo .zip que descargamos anteriormente y procedemos a extraer la carpeta del archivo comprimido.




Regresamos a Laragon y automáticamente detectará la nueva versión de PHP, solo hay que dar clic en la nueva versión.

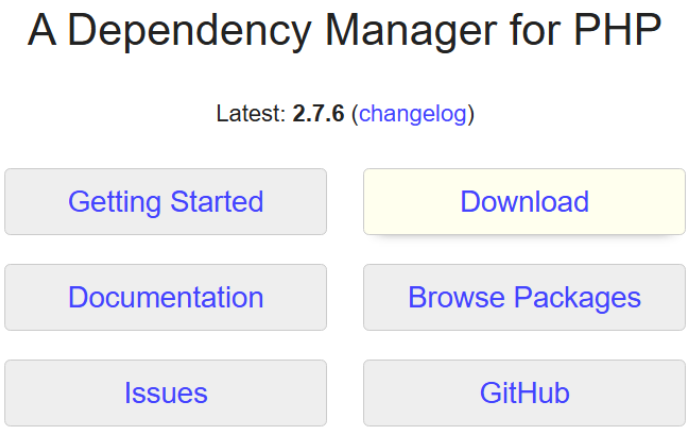



Finalmente, para comprobar que todo funciona, deberemos dar clic en iniciar todo, y si no se produce ninguna ventana de error todo funciona correctamente.

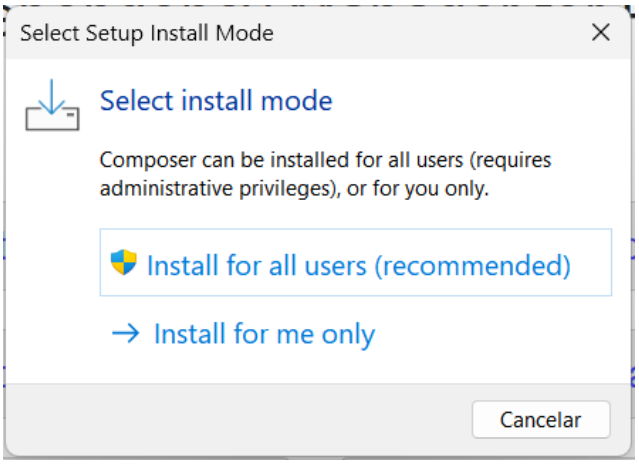



- Composer

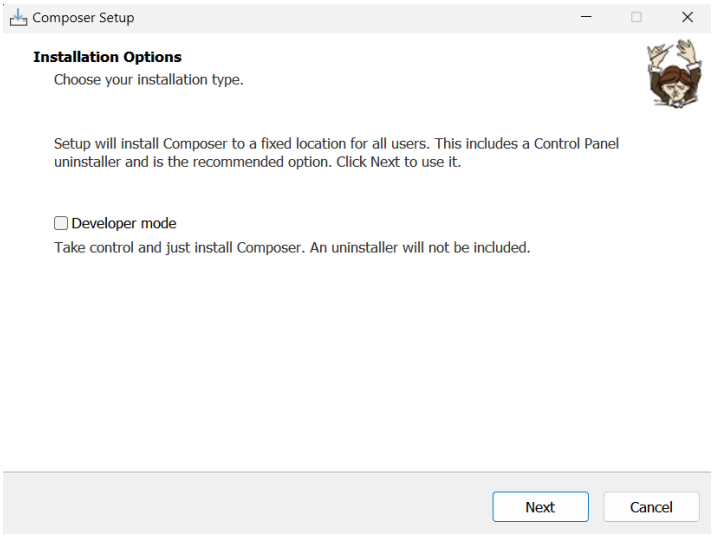
Finalmente, para manejar los diferentes paquetes de PHP se usa una herramienta llamada Composer, similar a lo que hace npm en Node JS. Para instalarla basta con ir a su página oficial "https://getcomposer.org/" . Y dar clic en el botón **Download** y descargar el archivo ejecutable.



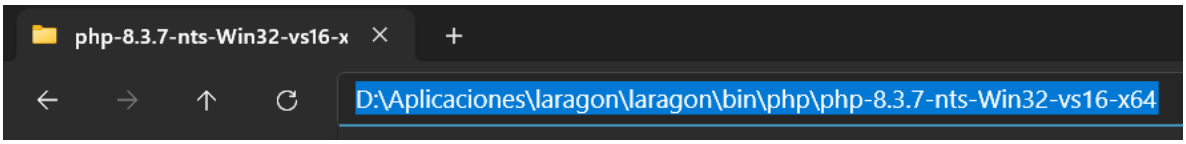

Al ejecutar el instalador veremos la siguiente ventana, en donde se menciona si deseamos instalar Composer para todos los usuarios creados en la computadora, o únicamente para el usuario actual. **Esta elección queda a consideración del administrador y el protocolo de seguridad que se siga**. En nuestro caso seleccionamos **Install for all users.**



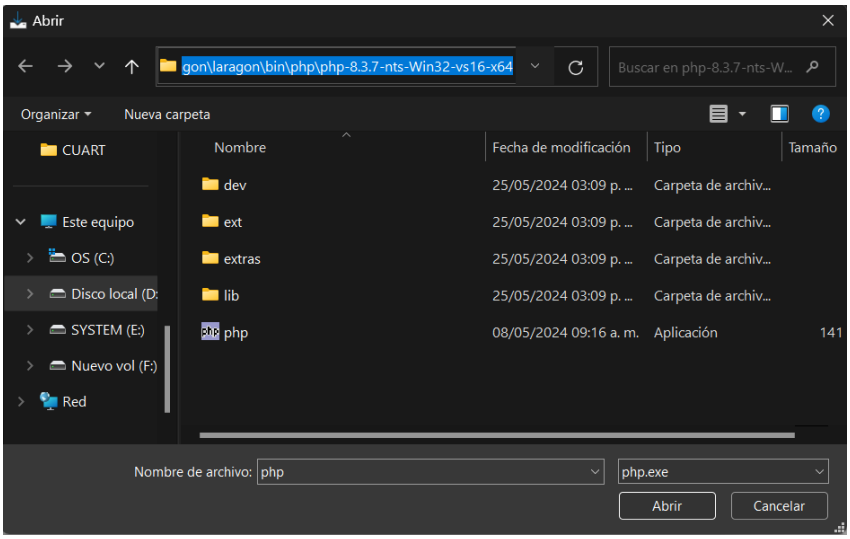

Posteriormente se muestra una ventana que nos permite activar la instalación en modo desarrollador, en este caso no la activaremos y daremos clic en **Next**.



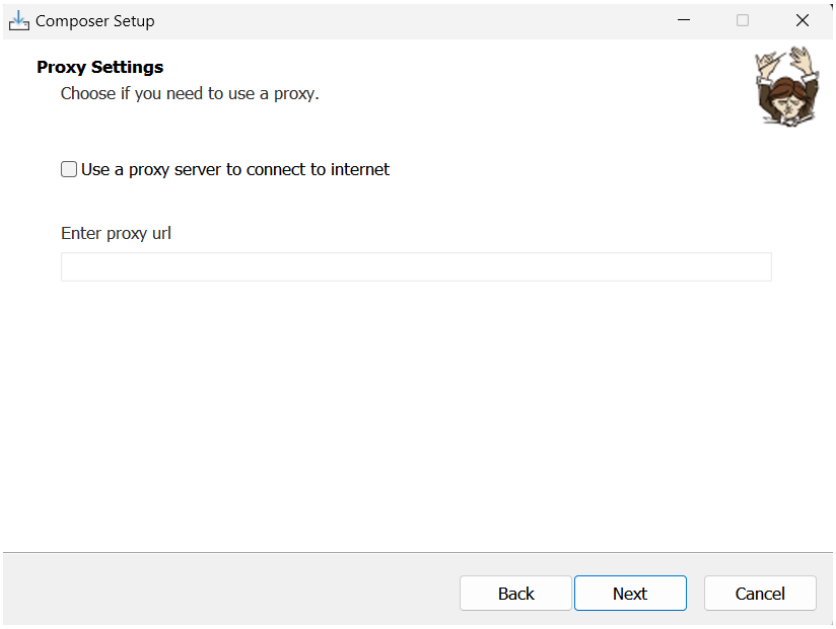

Antes de continuar hay volver a la ruta donde descomprimimos la nueva versión de PHP. Copiamos esa ruta:



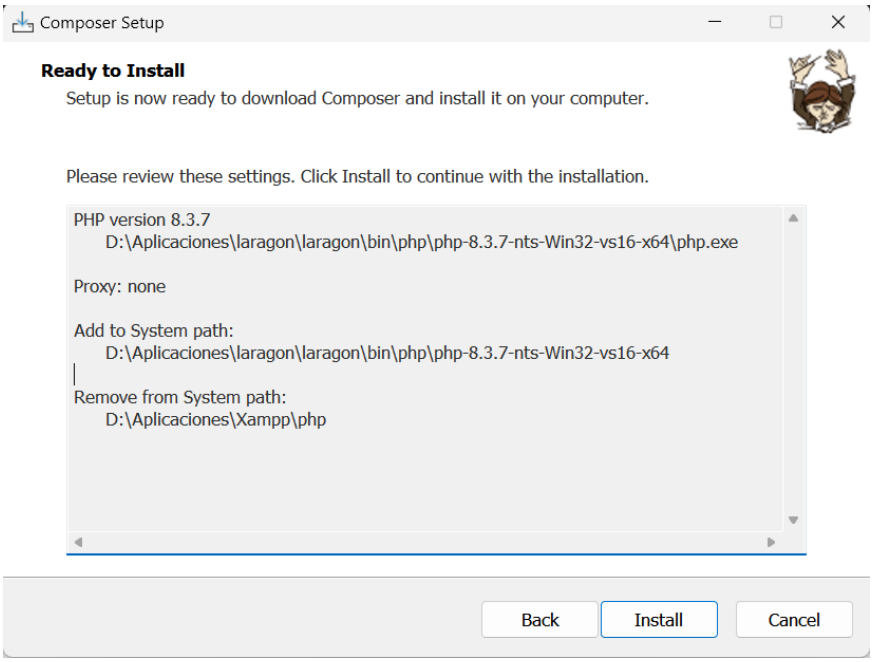

El siguiente paso es seleccionar el archivo .exe de PHP, damos clic en el botón **Browse...** y en el explorador de archivos colocamos la ruta que copiamos. Nos dirigirá a la carpeta de PHP, y dentro seleccionaremos el archivo llamado **php** y damos clic en abrir



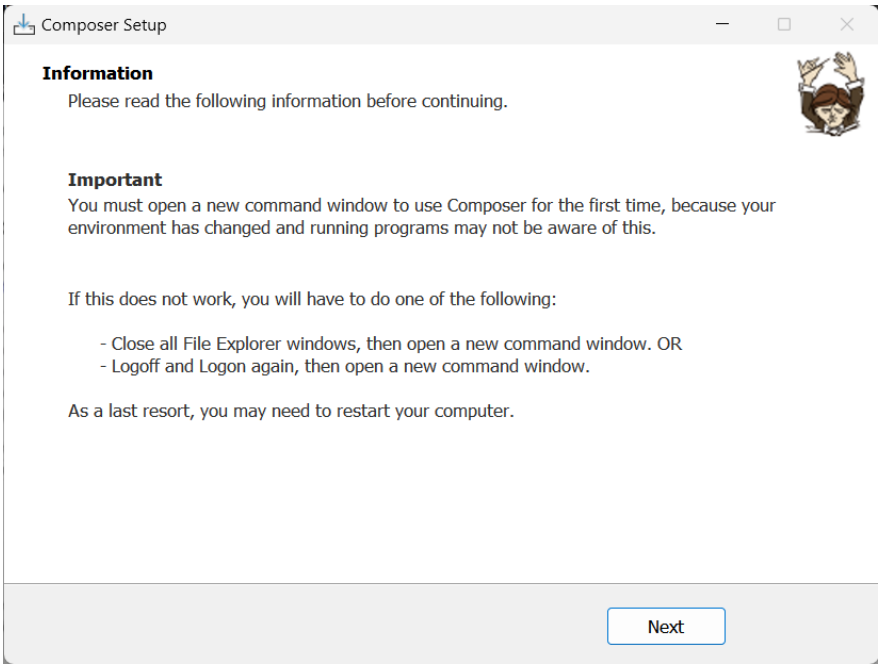

Una vez seleccionada la versión adecuada damos clic en **Next.** Y el instalador verificará que la ruta que se proporcionó esta correcta. Una vez comprobada podremos configurar un proxy, para el funcionamiento del proyecto no es necesario. Así que damos clic en **Next**



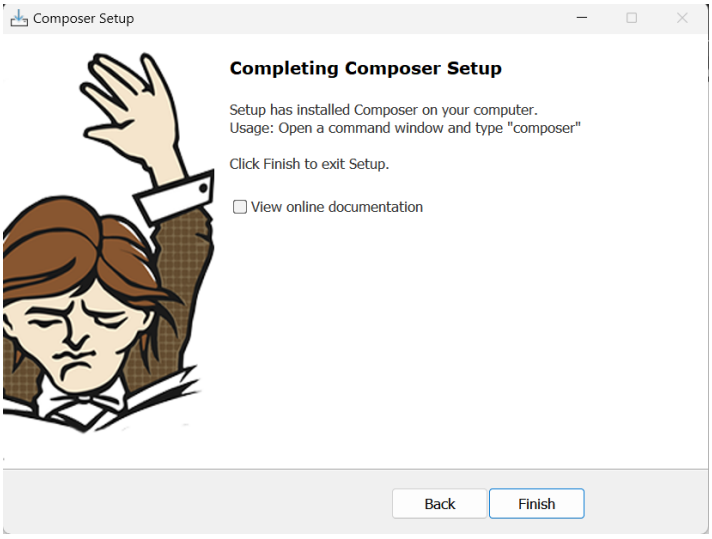
Confirmamos los datos y damos clic en **Install** y comenzará a instalarse




Al terminar nos mostrará un aviso sobre algunos problemas que podríamos tener, en donde como solución nos recomiendan abrir una nueva ventana de comando o reiniciar la computadora, damos clic en **Next**



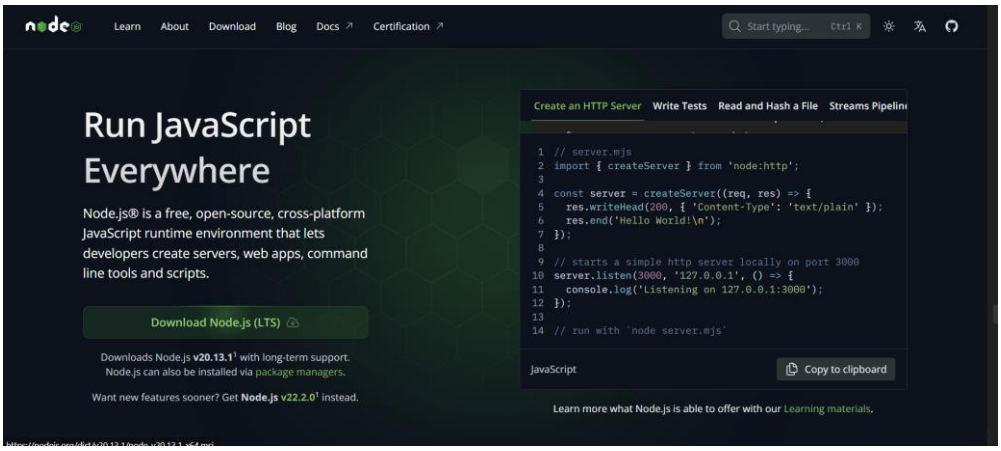
Para finalizar nos muestra un mensaje de confirmación y damos clic en **Finish**



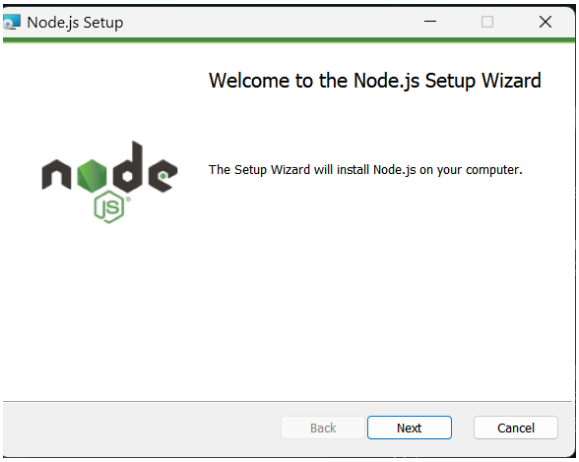

Para comprobar que Composer se instaló correctamente, abrimos una nueva ventana de comandos y en la línea de comandos escribimos **composer**, y nos debería mostrar la versión actual


- Angular
    - Node JS

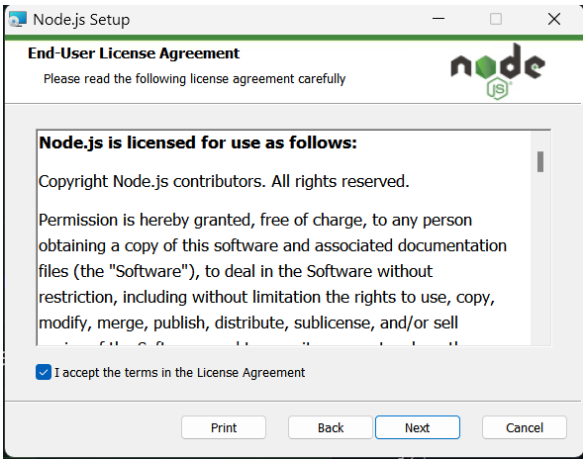
Para poder instalar Angular necesitaremos de Node JS, que es el entorno de desarrollo utilizado por Angular. Para instalarlo basta con entrar a su página oficial "https://nodejs.org/en/" y descargar la versión más reciente



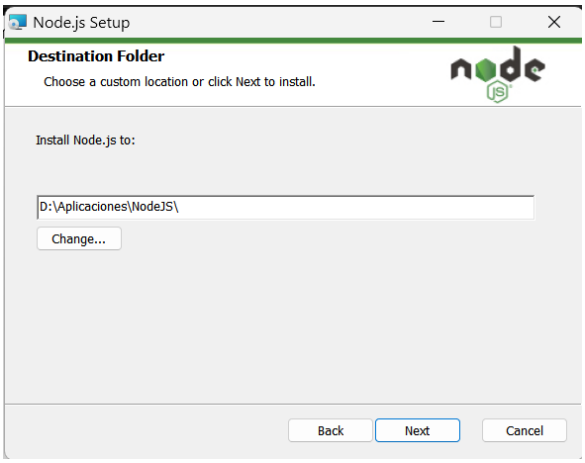

Al ejecutar el archivo .exe se mostrará una nueva ventana y el programa calculará el espacio disponible que tenemos en el disco duro, si existe suficiente espacio se activará el botón **Next**

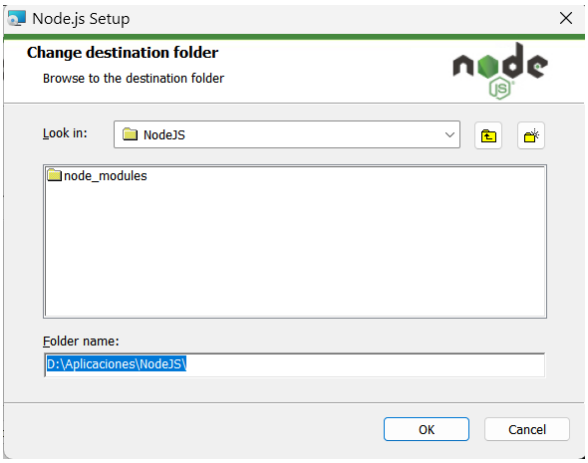



Aceptamos los términos y condiciones, y damos clic en Next

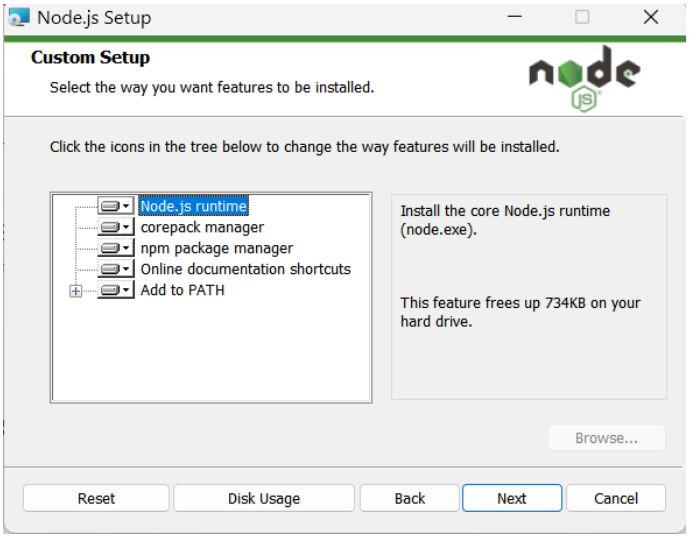


A continuación, podemos configurar la ruta donde se instalará el programa, por defecto ya se crea una ruta, pero si se desea modificar, hacemos clic en el botón **Change…** y buscamos la ruta en la ventana que se abre

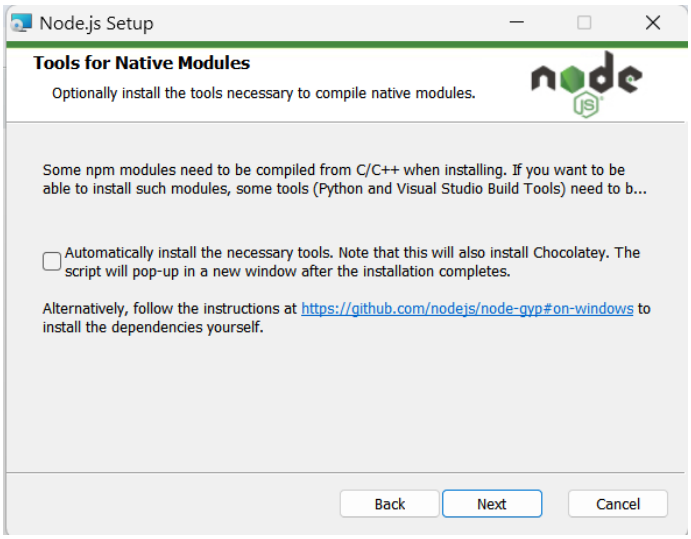


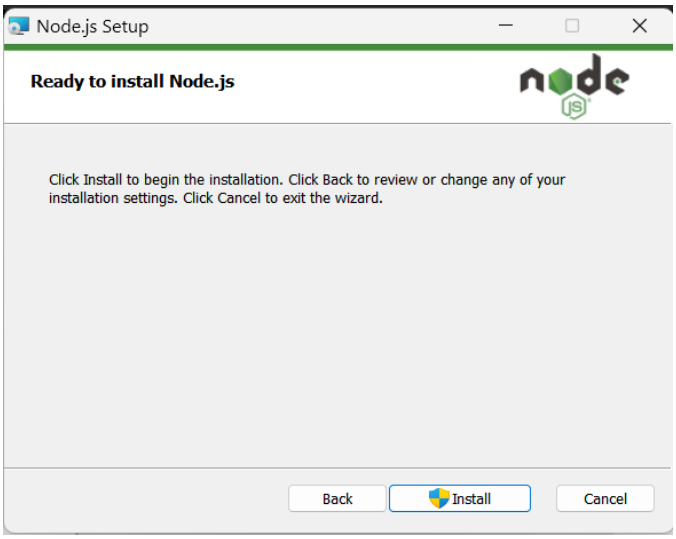

Podremos elegir que caracteristicas instalar, sin embargo, no es necesario cambiar alguna configuración, por lo que solo presionamos Next



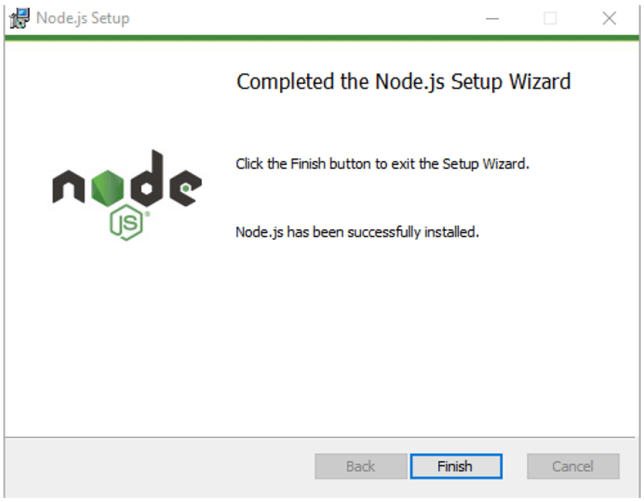

Posteriormente Node indica que algunos modulos necesitan de herramientas adicionales para funcionar, en esta ventana podemos seleccionar si instalarlas o no, en nuestro caso no las instalaremos, así que dejamos la casilla deseleccionada y presionamos continuar



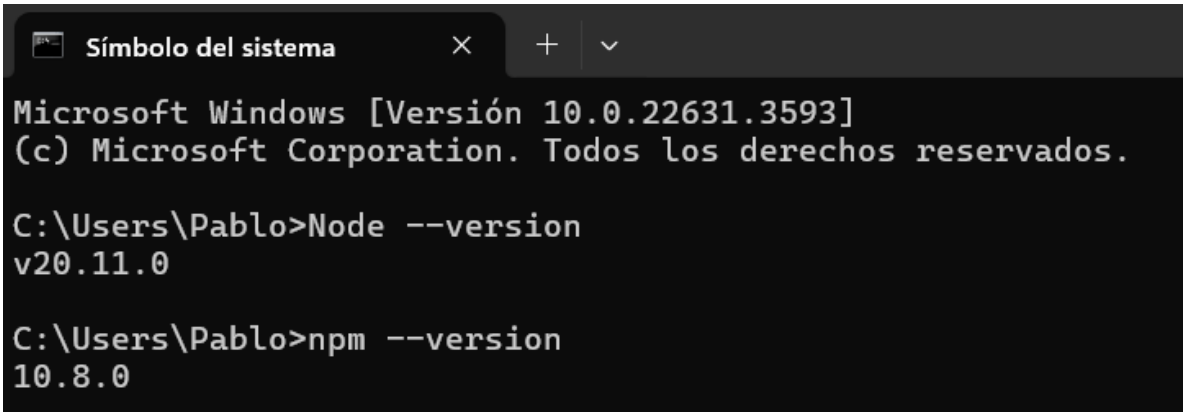

Finalmente confirmamos la instalacion dando clic en el boton **Install**




Una vez terminada la instalación damos clic en el botón de **Finish**

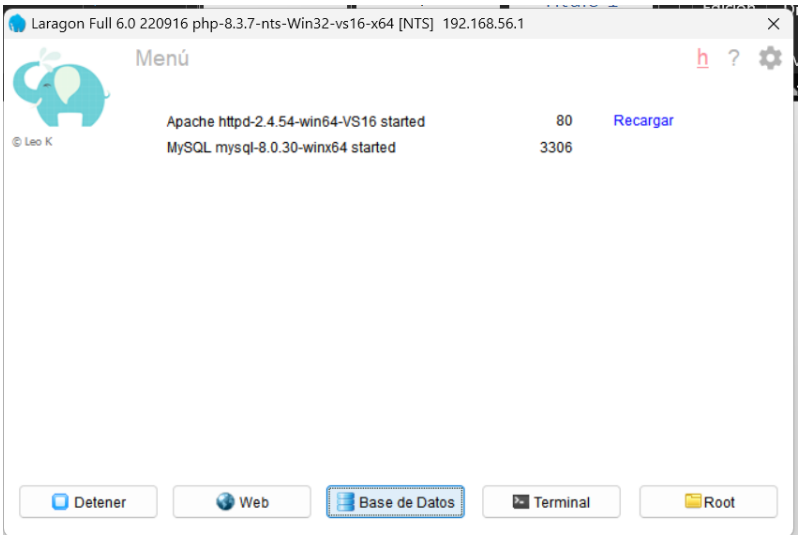



Finalmente para comprobar que se instalo correctamente, abrimos una ventana de comandos y colocamos los comandos **Node** --**version** y **npm** --**version,** si todo se instalo correctamente deberiamos ver la version de Node y de npm

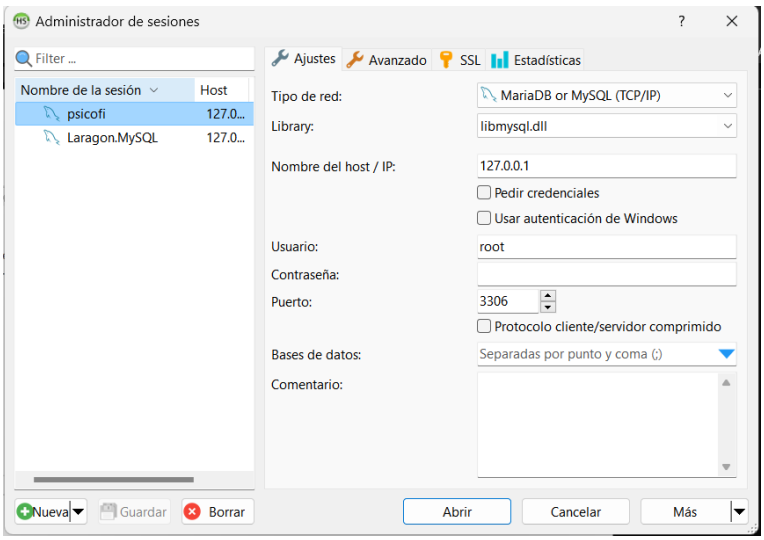



- MySQL

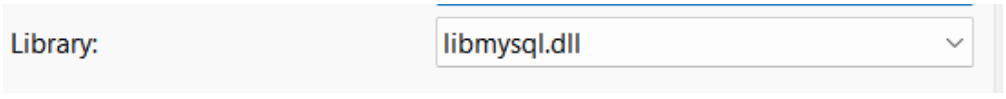
Al instalar Laragon se instalo al mismo tiempo MySQL, por lo que lo unico que debemos hacer es crear la BD, para ello iniciaremos Laragon y daremos clic en el botón **Base de datos**




Se nos mostrará una nueva ventana, en donde darmos clic en botón **Nueva** y al registro que nos creo le daremos el nombre de **psicofi**




Finalmente dentro de los ajustes cambiaremos el campo **Library** por el valor **libmysql.dll**

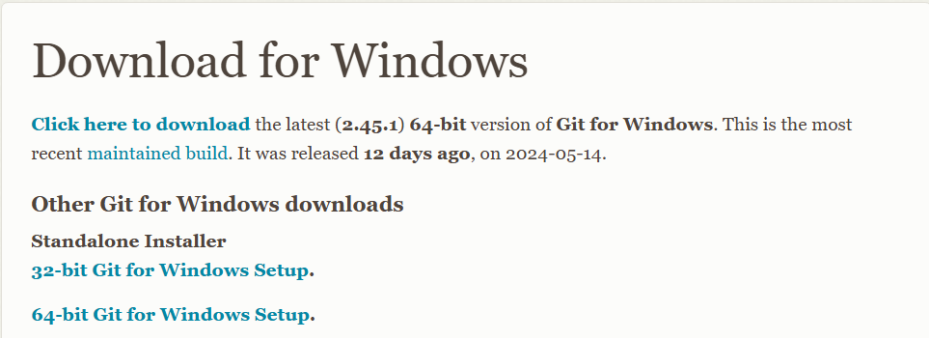


- Git

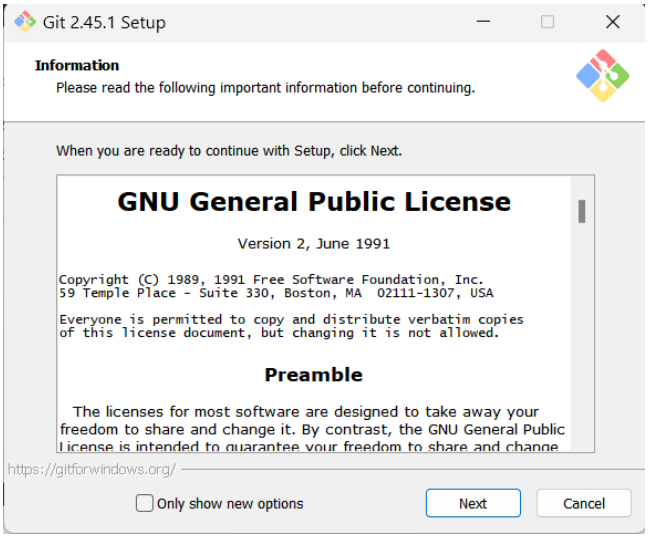
Git nos permitirá obtener la información del proyecto desde el repositorio de github, para instalar GIT necesitamos descargar el instalador desde su página oficial "https://git-scm.com/" en el apartado **Downloads** y seleccionando la verisón para el sistema operativo que se tenga instalado.



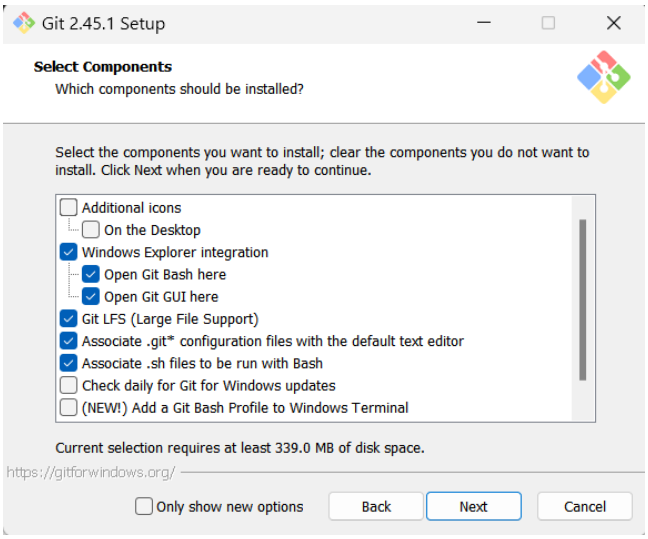
En este caso se uso Windows. Así que descargamos la versión que corresponda con la arqutiectura de la computadora (32 o 64 bits) y ejecutamos el instalador.



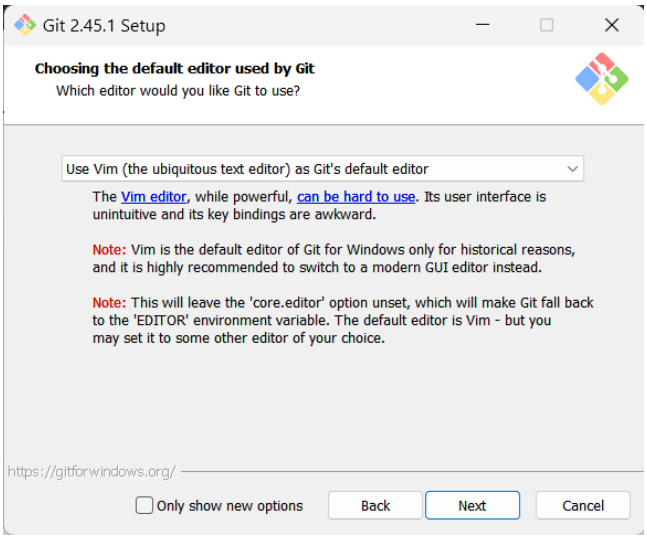
Al abrir el instalador lo primero que encontraremos serán los terminos y condiciones, damos clic en **Next**

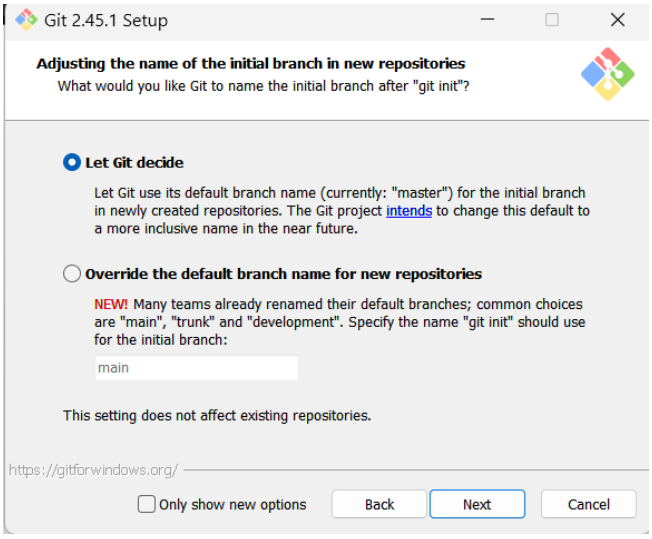


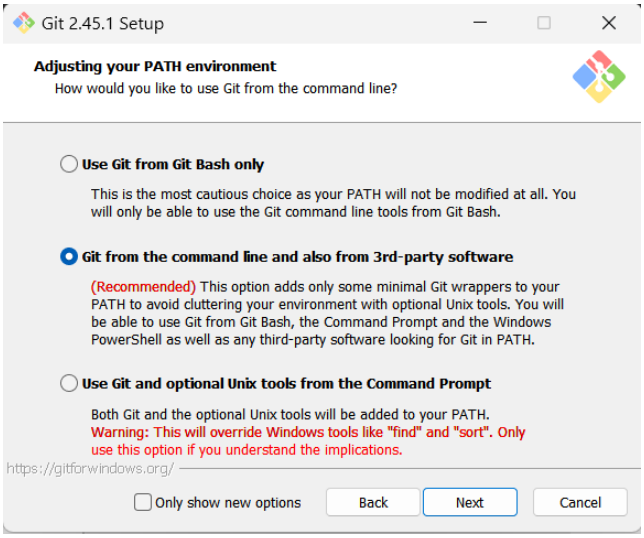

Podremos configurar los componentes que deseamos instalar, no es necesario modificar nada, así que podemos continuar

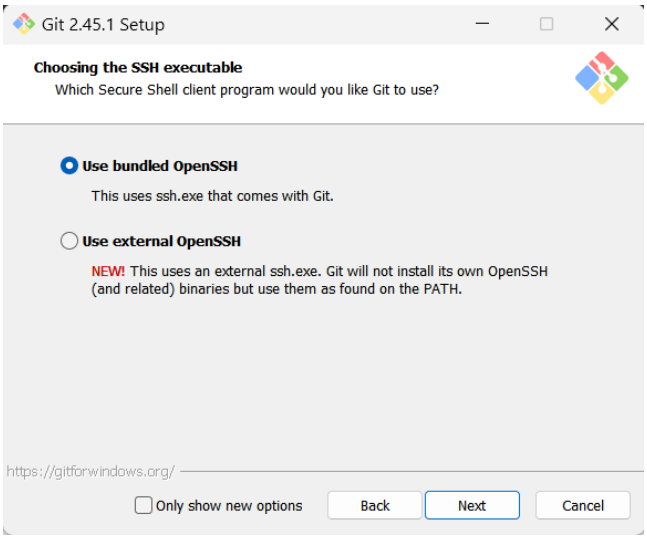


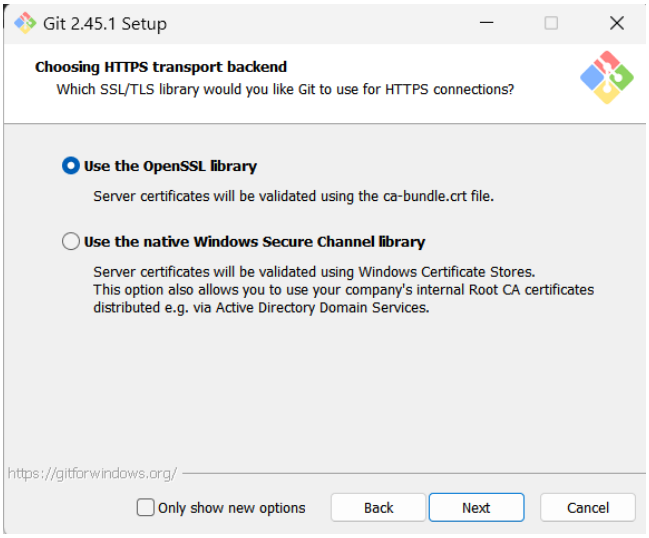

Así mismo podremos elegir un editor de texto para git. Este paso y los siguientes quedan a elección del administrador, pues en su mayoria son elecciones personales acerca de como se inicializan los repositorios y el manejo de los mismos. Recomendamos dejar las opciones por defecto, pues para este proyecto se uso la configuración por defecto.

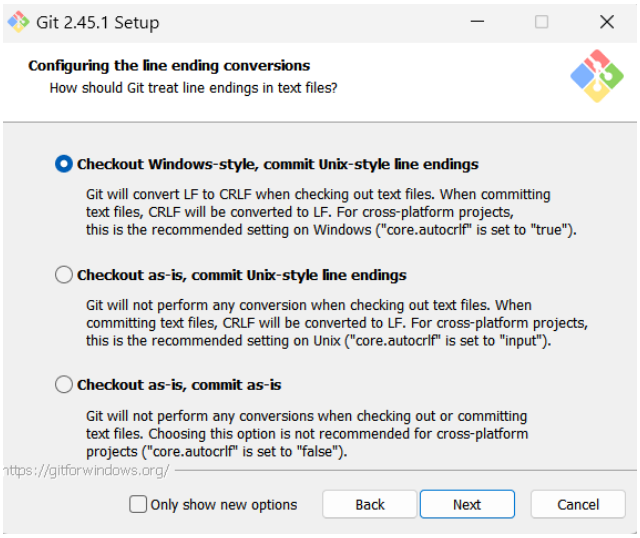

Git 2.45.1 Setup
Choosing the default editor used by Git
Which editor would you like Git to use?
Use Vim (the ubiquitous text editor) as Git's default editor
The Vim editor, while powerful, can be hard to use. Its user interface is unintuitive and its key bindings are awkward.
Note: Vim is the default editor of Git for Windows only for historical reasons, and it is highly recommended to switch to a modern GUI editor instead.
Note: This will leave the 'core.editor' option unset, which will make Git fall back to the 'EDITOR' environment variable. The default editor is Vim - but you may set it to some other editor of your choice.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

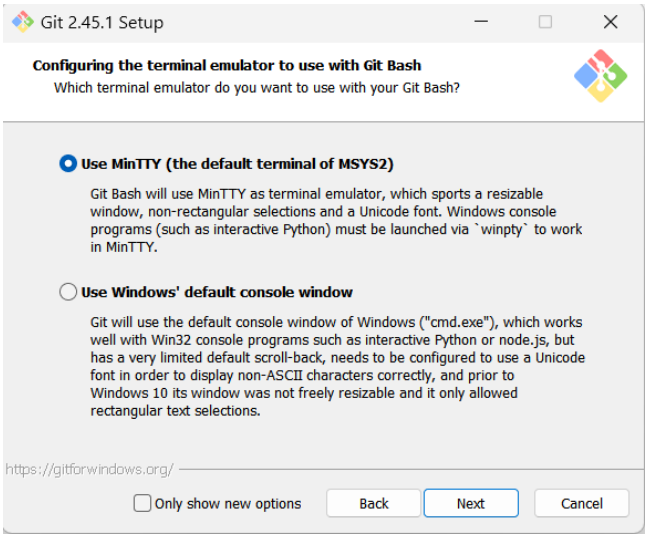

Git 2.45.1 Setup
Adjusting the name of the initial branch in new repositories
What would you like Git to name the initial branch after "git init"?
Let Git decide
Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project intends to change this default to a more inclusive name in the near future.
Override the default branch name for new repositories
NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:
main
This setting does not affect existing repositories.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

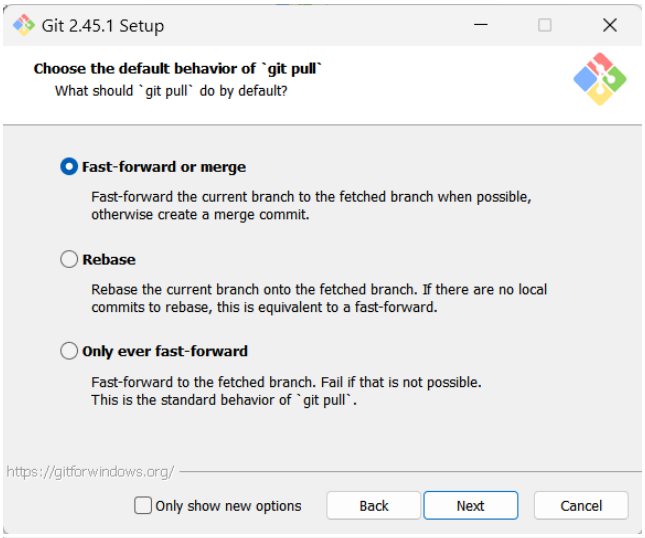

Git 2.45.1 Setup
Adjusting your PATH environment
How would you like to use Git from the command line?
Use Git from Git Bash only
This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.
Git from the command line and also from 3rd-party software
(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.
Use Git and optional Unix tools from the Command Prompt
Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

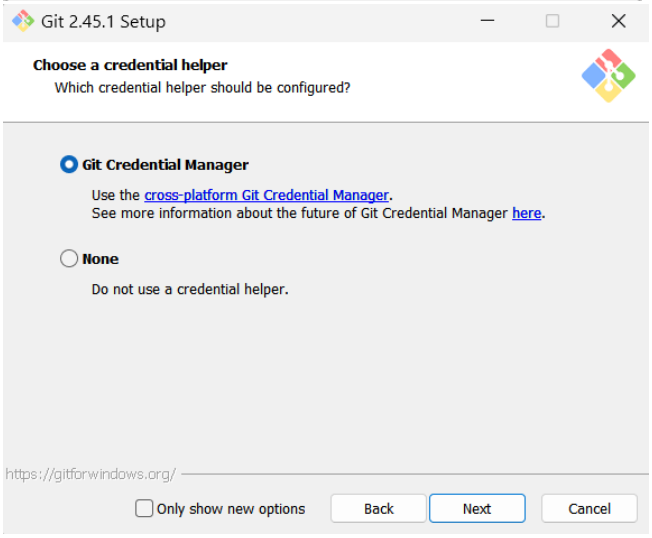

Git 2.45.1 Setup
Choosing the SSH executable
Which Secure Shell client program would you like Git to use?
Use bundled OpenSSH
This uses ssh.exe that comes with Git.
Use external OpenSSH
NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

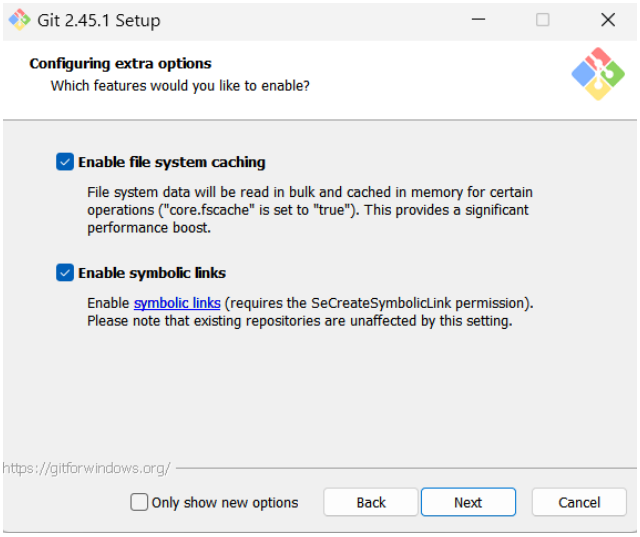

Git 2.45.1 Setup
Choosing HTTPS transport backend
Which SSL/TLS library would you like Git to use for HTTPS connections?
Use the OpenSSL library
Server certificates will be validated using the ca-bundle.crt file.
Use the native Windows Secure Channel library
Server certificates will be validated using Windows Certificate Stores.
This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel


Git 2.45.1 Setup
Configuring the line ending conversions
How should Git treat line endings in text files?
Checkout Windows-style, commit Unix-style line endings
Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").
Checkout as-is, commit Unix-style line endings
Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").
Checkout as-is, commit as-is
Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

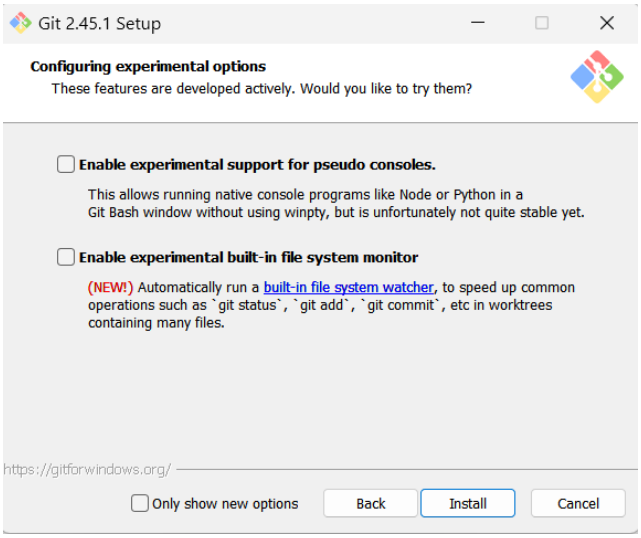

Git 2.45.1 Setup
Configuring the terminal emulator to use with Git Bash
Which terminal emulator do you want to use with your Git Bash?
Use MinTTY (the default terminal of MSYS2)
Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.
Use Windows' default console window
Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel


Git 2.45.1 Setup
Choose the default behavior of `git pull`
What should `git pull` do by default?
Fast-forward or merge
Fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.
Rebase
Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.
Only ever fast-forward
Fast-forward to the fetched branch. Fail if that is not possible. This is the standard behavior of `git pull`.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel

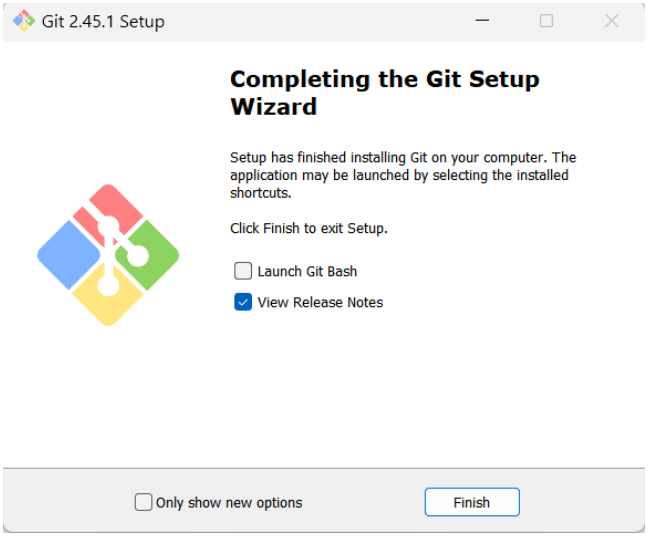

Git 2.45.1 Setup
Choose a credential helper
Which credential helper should be configured?
Git Credential Manager
Use the cross-platform Git Credential Manager.
See more information about the future of Git Credential Manager here.
None
Do not use a credential helper.
https://gitforwindows.org/
Only show new options
Back
Next
Cancel




Al llegar a la ultima ventana antes de instalar, podremos elegir instalar configuraciones experimentales. Como su nombre lo indican estan en proceso de prueba, por lo que para que todo funcione bien lo mejor es no instalarlas y damos clic en el botón Install




Al terminar la instalación se mostrará una ventana confirmando la instalación

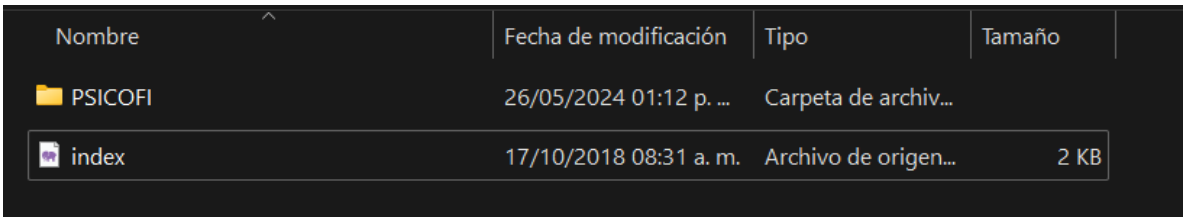



Instalación del proyecto

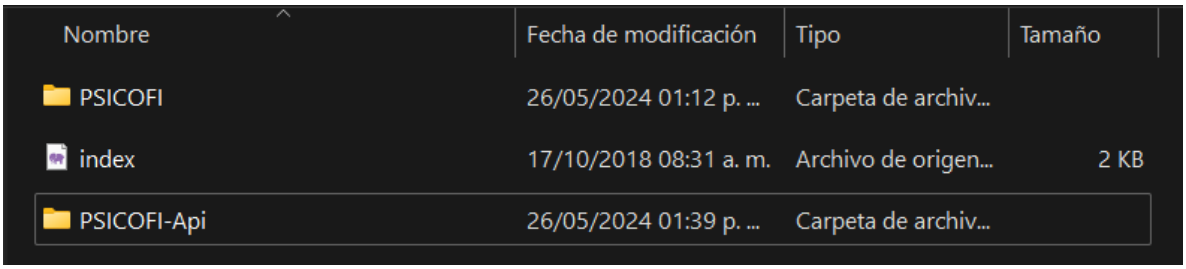
Clonar repositorio

Una vez que se instalaron todas las herramientas necesitamos descargar el proyecto desde el repositorio de github “https://github.com/FernandoRmzMtz/PSICOFI.git”. Antes de clonar el repositorio es necesario ubicar la carpeta donde se instalara el proyecto, para ello recomendamos ubicar la carpeta www de laragon, esta se encuentra en la ruta de instalación de laragon . En ella abriremos una nueva ventana de comandos y colocaremos el comando git clone https://github.com/FernandoRmzMtz/PSICOFI.git, el cual descargara todo el proyecto del repositorio

```
PS D:\Universidad\PSICOFI> git clone https://github.com/FernandoRmzMtz/PSICOFI.git
Cloning into 'PSICOFI'...
remote: Enumerating objects: 2935, done.
remote: Counting objects: 100% (1022/1022), done.
remote: Compressing objects: 100% (505/505), done.
remote: Total 2935 (delta 715), reused 705 (delta 513), pack-reused 1913
Receiving objects: 100% (2935/2935), 11.73 MiB | 8.92 MiB/s, done.
Resolving deltas: 100% (1946/1946), done.
```



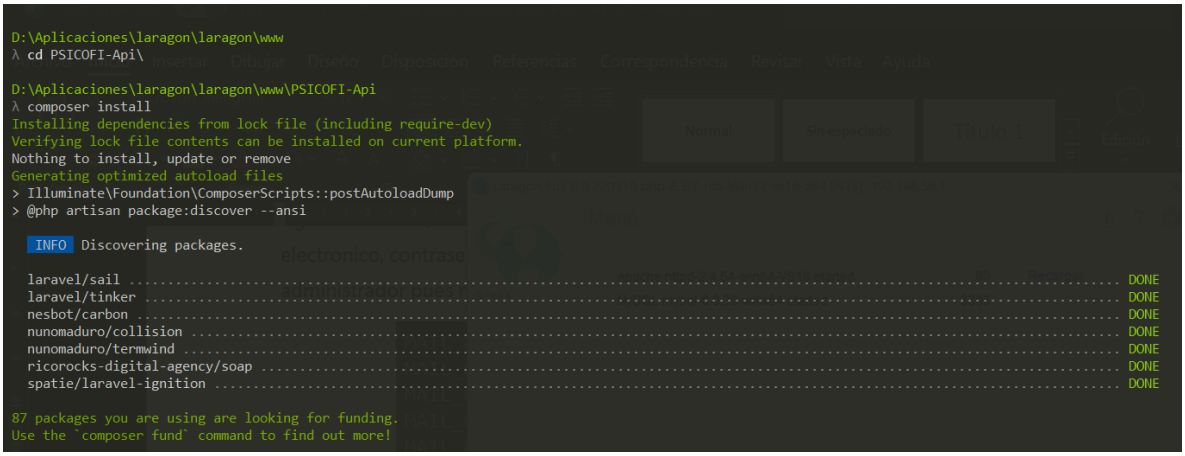

Dentro encontraremos tres carpetas, de las cuales la carpeta PSICOFI-Web corresponde al código de Angular y PSICOFI-Api al código de Laravel. Copiaremos la carpeta PSICOFI-Api y la pegaremos fuera de la carpeta PSICOFI, es decir, en la carpeta www




A continuación se muestran los pasos necesarios para hacer funcionar cada uno de los módulos de la plataforma.

- Laravel (PSICOFI-Api)

Para comenzar iniciaremos Laragon para encender el servidor local. Desde el mismo laragon damos clic en el botón Terminal nos abrira una nueva ventana con una terminal. Cambiaremos el directorio a la carpeta de PSICOFI-Api y colocaremos el comando composer install, este comando se encarga de instalar todas las dependencias de laravel, puede que este proceso tarde un poco.




Posteriormente debemos configurar el archivo .env de laravel. Dentro de la carpteta PISCOFI-Api encontraremos un archivo .env.example, lo duplicaremos y le cambiaremos el nombre a .env y lo abriremos con un editor de texto como notepad++ o el bloc de notas

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=laravel
DB_USERNAME=root
DB_PASSWORD=
```

En este archivo cambiaremos las siguientes lineas de código.

DB_DATABASE. Esta linea se encarga de definir con que base de datos se va a conectar, como nosotros definimos la base de datos psicofi anteriormente cambiaremos el nombre de laravel a psicofi.

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=psicofi
DB_USERNAME=root
DB_PASSWORD=
```

Opcionalmente se recomienda configurar el DB_USERNAME y DB_PASSWORD, debido a que el root solo suele ser utilizado en un entorno local, en producción es necesario hacer los ajustes para evitar vulnerabilidad de seguridad, en lugar de root, se debe crear un usuario especifico con su contraseña cifrada y así asegurar accesos no autorizados.

En el proyecto se hace uso de un servicio de correos electronicos. Para configurar la cuenta desde la cuál se enviarán estos mensajes se cambian las siguientes líneas, en donde se define, el protocolo a utilizar, la cuenta de correo electronico, contraseña, entre otros, esta información es configurada por el administrador pues necesita de información sensible.

```
MAIL_MAILER=smtp
MAIL_HOST=mailpit
MAIL_PORT=1025
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
MAIL_FROM_ADDRESS="hello@example.com"
MAIL_FROM_NAME="${APP_NAME}"
```

Al terminar ejecutaremos el comando php artisan key:generate.

```
D:\Aplicaciones\laragon\laragon\www\PSICOFI-Api
λ php artisan key:generate

  INFO  Application key set successfully.
```

Finalmente para comprobar que todo esta correcto, volvemos a la ventana de comandos y ejecutamos los comandos php artisan config:clear para que la aplicación vuelva a cargar el archivo .env que modificamos y posteriormente el comando php artisan migrate:fresh que creará las tablas de la BD

```
D:\Aplicaciones\laragon\laragon\www\PSICOFI-Api
λ php artisan config:clear

  INFO  Configuration cache cleared successfully.


D:\Aplicaciones\laragon\laragon\www\PSICOFI-Api
λ php artisan migrate:fresh

  Dropping all tables ..... 1,307ms DONE

  INFO  Preparing database.

  Creating migration table ..... 267ms DONE

  INFO  Running migrations.

  2024_02_24_010623_psicologo_externo ..... 295ms DONE
  2024_02_24_010624_estado_cita ..... 221ms DONE
  2024_02_24_010625_tipo_intervencion ..... 187ms DONE
  2024_02_24_010626_alumno ..... 221ms DONE
  2024_02_24_023235_administrador ..... 199ms DONE
  2024_02_24_024311_carreras_psico ..... 199ms DONE
  2024_02_24_024312_psicologo ..... 471ms DONE
  2024_02_24_041308_cita ..... 1,302ms DONE
  2024_02_24_051435_nota_cita ..... 522ms DONE
  2024_02_24_052710_historial ..... 566ms DONE
  2024_02_24_053437_reporte ..... 608ms DONE
  2024_02_24_054551_agenda ..... 859ms DONE
  2024_05_13_150757_departamento_table ..... 132ms DONE
```

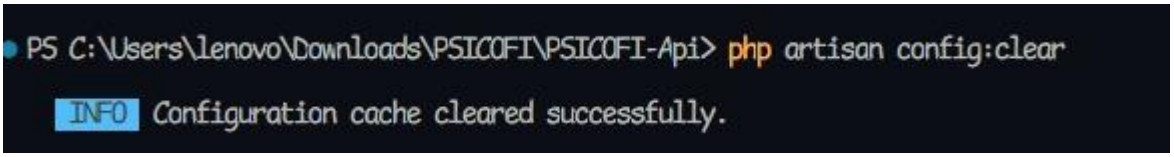
Todos estos comandos deben ejecutarse solo la primera vez que se instala la BD.

Después de configurar el archivo .env y ejecutar los comandos iniciales, se debe iniciar un servidor de desarrollo local con el siguiente comando:


Este comando inicia un servidor web ligero que permite probar la aplicación en un navegador. Por defecto, estará disponible en http://127.0.0.1:8000

Para asegurar que las rutas están configuradas correctamente, se pueden listar las rutas disponibles ejecutando:


Si se necesita realizar cambios en el archivo .env después de configurar el proyecto, asegurar limpiar la caché de configuración con




Los comandos mencionados para configurar la base de datos y generar la clave solo deben ejecutarse la primera vez que instalas el proyecto. En futuras ocasiones, solo necesitas iniciar Laragon y ejecutar php artisan serve para levantar el servidor.

- Angular (PSICOFI-Web)

Para la inicialización de la página web lo que necesitamos es entrar en la carpeta PSICOFI y posteriormente abrir la carpeta de **PSICOFI-Web**, donde iniciaremos una nueva consola de comandos y colocaremos el comando **npm install --force** lo que instalará las dependencias del proyecto

```
PS D:\Aplicaciones\laragon\laragon\www\PSICOFI\PSICOFI-Web> npm install --force
npm warn using --force Recommended protections disabled.
npm warn ERESOLVE overriding peer dependency
npm warn While resolving: igniteui-angular@17.1.1
npm warn Found: @angular/animations@16.2.12
npm warn node_modules/@angular/animations
npm warn   @angular/animations@"^16.2.12" from the root project
npm warn   1 more (@angular/platform-browser)
```

Una vez que se terminan de instalar las dependencias, hay que ejecutar el siguiente comando ng serve, para iniciar la página web, este proceso puede tardar un poco y debe realizarse cada que la página quede fuera de servicio o necesite reiniciarse, además cabe mencionar que para que todo funcione correctamente, Laragon debe estar encendido al mismo tiempo que se ejecuta la página web.

Si todo fue bien, deberíamos poder ver la pantalla de inicio de la página web.