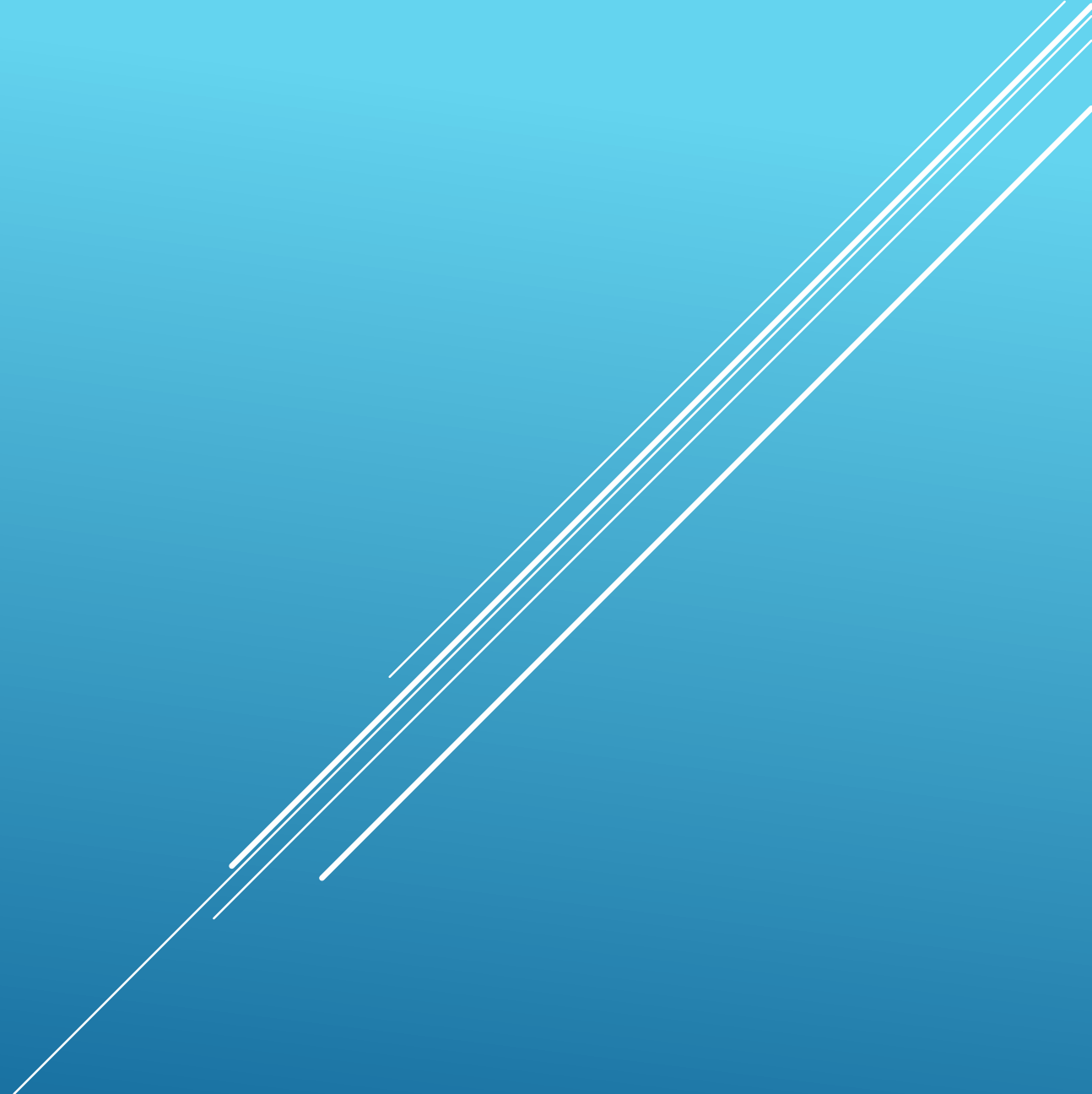


# SYMPHONY 7

INSTALACIÓN



# HERRAMIENTAS QUE SE VAN A UTILIZAR

Un framework, es una estructura base predefinida donde desarrollar una aplicación, para poder utilizar el framework de symfony, necesitamos:

- Gestor de dependencias:

Existen muchas herramientas de gestión de dependencias, aquí vamos a utilizar composer.

- PHP superior a la versión 8.1

Como vamos a utilizar el cmd para lanzar instrucciones PHP, debemos configurar las variables de entorno.

# CONFIGURAR LAS VARIABLES DE ENTORNO

Las variables de entorno definen las configuraciones predeterminadas de cada programa.

En Windows se encuentran en :

Panel de control->Sistema y seguridad->sistema:

A la izquierda de esta pantalla se encuentra la opción de:

- configuración avanzada del sistema



Sistema



Panel de control > Sistema y seguridad > Sistema



Buscar en el Panel de control



Ventana principal del Panel de control



Administrador de dispositivos



Configuración de Acceso remoto



Protección del sistema



[Configuración avanzada del sistema](#)

## Ver información básica acerca del equipo

### Edición de Windows

Windows 10 Pro

© 2017 Microsoft Corporation. Todos los derechos reservados.



# Windows 10

### Sistema

Procesador: Intel(R) Core(TM) i7-5500U CPU @ 2.40GHz 2.39 GHz  
Memoria instalada (RAM): 2,00 GB  
Tipo de sistema: Sistema operativo de 64 bits, procesador x64  
Lápiz y entrada táctil: La entrada táctil o manuscrita no está disponible para esta pantalla

### Configuración de nombre, dominio y grupo de trabajo del equipo

Nombre de equipo: DESKTOP-QSQTDRO  
Nombre completo de equipo: DESKTOP-QSQTDRO  
Descripción del equipo:  
Grupo de trabajo: WORKGROUP



[Cambiar configuración](#)

### Activación de Windows

Windows no está activado. [Lea los Términos de licencia del software de Microsoft](#)

Id. del producto: 00331-10000-00001-AA681



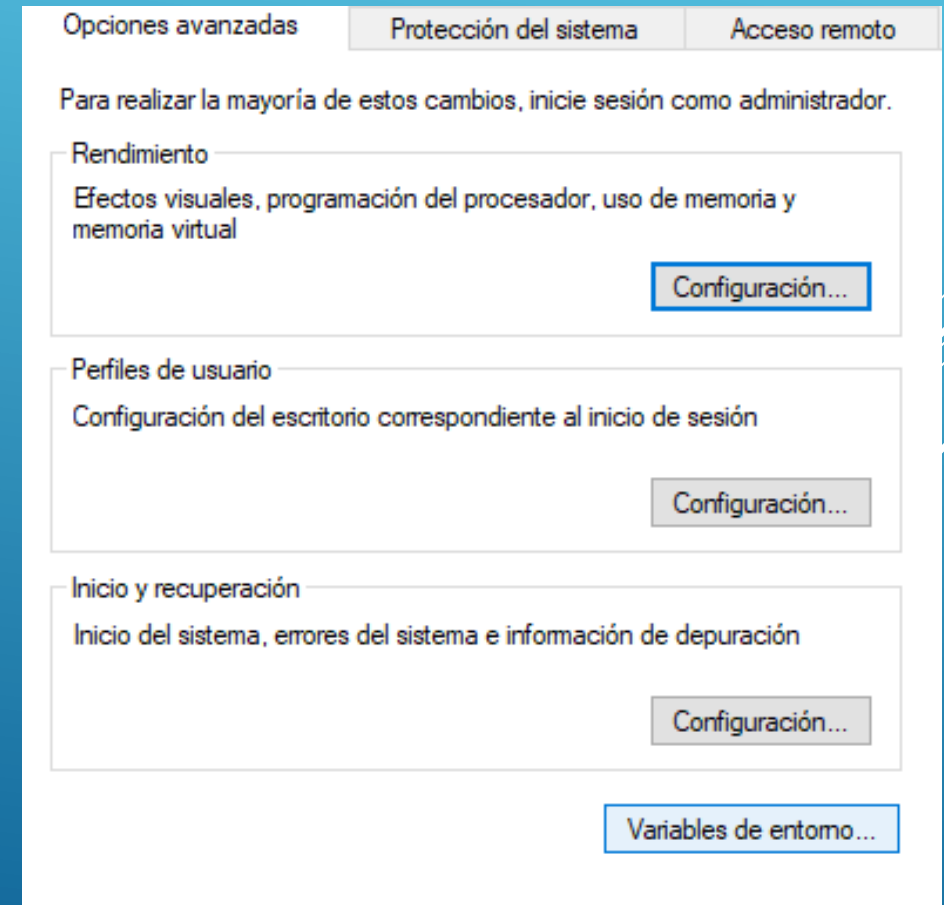
[Activar Windows](#)

Vea también

Seguridad y mantenimiento

# CONFIGURAR LAS VARIABLES DE ENTORNO

Para que el ejecutable este en todas las carpetas  
haremos click en variables de entorno:



# CONFIGURAR LAS VARIABLES DE ENTORNO

Y en el apartado de variables del sistema, nos desplazamos hasta la variable path, haciendo doble click sobre ella.

Aquí podremos agregar a los caminos ya presentes, la ruta hacia el ejecutable de php.

## **Ubicación del ejecutable de php:**

En C:\wamp64\bin\php\php8.2.13

Copiamos esta dirección y volvemos al panel, hacemos click en nuevo en la variable de entorno path , pegamos la ruta que hemos copiado, y hacemos click en ok, para después cerrar los paneles.

## Variables de entorno

## Variables de usuario para alumno

Variable	Valor
OneDrive	C:\Users\alumno\OneDrive
Path	C:\Users\alumno\AppData\Local\Microsoft\WindowsApps;C:\User...
TEMP	C:\Users\alumno\AppData\Local\Temp
TMP	C:\Users\alumno\AppData\Local\Temp

Nueva...

Editar...

Eliminar

## Variables del sistema

Variable	Valor
ComSpec	C:\Windows\system32\cmd.exe
NUMBER_OF_PROCESSORS	1
OS	Windows_NT
Path	C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Familly 6 Model 61 Stepping 4. GenuineIntel

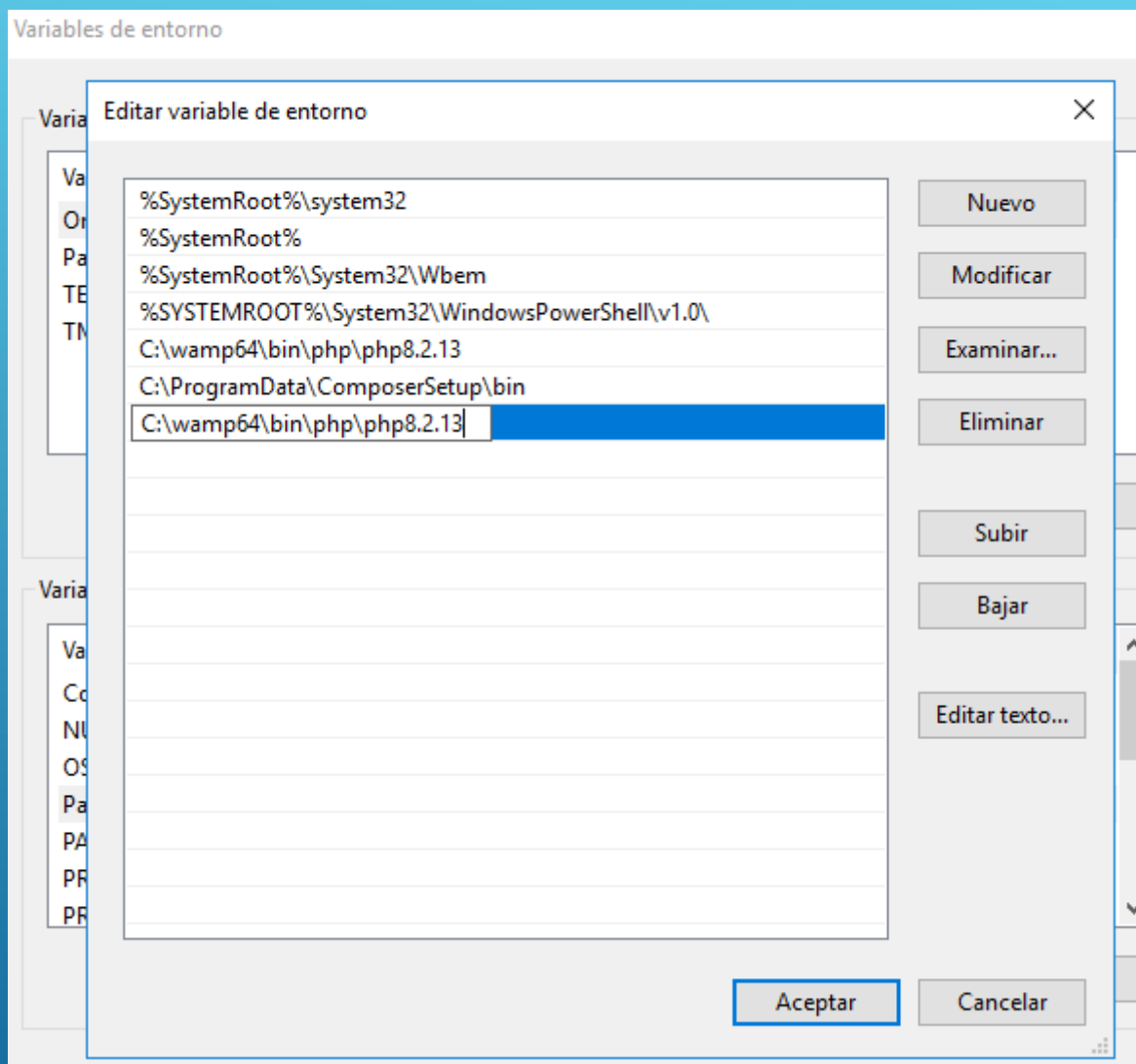
Nueva...

Editar...

Eliminar

Aceptar

Cancelar

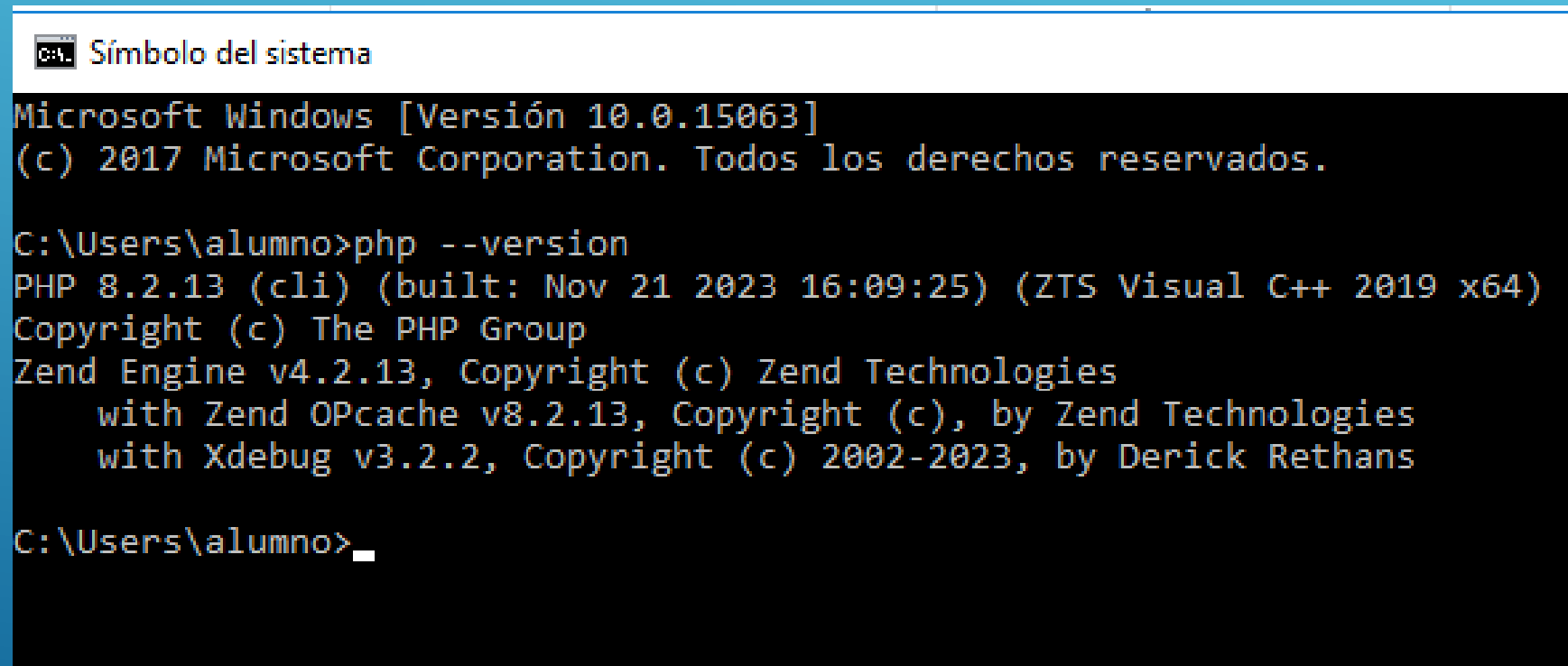




# CONFIGURAR LAS VARIABLES DE ENTORNO

Para comprobar que está correcto abrimos un terminal y escribimos por ejemplo:

`php --version.`

A screenshot of a Windows Command Prompt window. The title bar is white with a black icon and the text 'Símbolo del sistema'. The main area has a black background with white text. The text shows the Windows version (10.0.15063), copyright information (2017 Microsoft Corporation), the current directory (C:\Users\alumno), and the output of the 'php --version' command, which displays PHP 8.2.13 (cli) built on Nov 21 2023, along with details about the Zend Engine, OPcache, and Xdebug.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.15063]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\alumno>php --version
PHP 8.2.13 (cli) (built: Nov 21 2023 16:09:25) (ZTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.2.13, Copyright (c) Zend Technologies
    with Zend OPcache v8.2.13, Copyright (c), by Zend Technologies
    with Xdebug v3.2.2, Copyright (c) 2002-2023, by Derick Rethans

C:\Users\alumno>
```

# INSTALAR COMPOSER

Ahora que php está disponible instalaremos composer, vamos a la pagina de composer, hacemos click en Download. Nos mostrará los siguientes comandos:

## Download Composer Latest: v2.8.3

### Windows Installer

The installer - which requires that you have PHP already installed - will download Composer for you and set up your PATH environment variable so you can simply call `composer` from any directory.

Download and run [Composer-Setup.exe](#) - it will install the latest composer version whenever it is executed.

### Command-line installation

To quickly install Composer in the current directory, run the following script in your terminal. To automate the installation, use [the guide on installing Composer programmatically](#).

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413
php composer-setup.php
php -r 'unlink('composer-setup.php');'
```

This installer script will simply check some `php.ini` settings, warn you if they are set incorrectly, and then download the latest `composer.phar` in the current directory. The 4 lines above will, in order:

# INSTALAR COMPOSER

De todos ellos solo nos servirán estas líneas:

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"  
php composer-setup.php
```

Ahora desde el terminal seleccionamos la carpeta `c:/wamp64/www` (`cd c:/wamp64/www`), y a continuación ejecutamos las líneas anteriores con el terminal abierto como administrador( botón derecho sobre el icono de cmd y abrir como administrador)

# INSTALAR COMPOSER

Una vez instalado para comprobar que la instalación es correcta ejecuta:

**php composer.phar -h**

Y nos mostrará la lista de comandos disponibles en composer.

Several white diagonal lines of varying lengths and thicknesses are positioned in the bottom right corner of the slide, creating a modern, abstract graphic element.

Administrador: Símbolo del sistema

Microsoft Windows [Versión 10.0.15063]

(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>cd c:/wamp64/www

c:\wamp64\www>php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

c:\wamp64\www>php composer-setup.php

All settings correct for using Composer

Downloading...

Composer (version 2.8.3) successfully installed to: C:\wamp64\www\composer.phar

Use it: php composer.phar

c:\wamp64\www>php composer.phar -h

Description:

List commands

Usage:

list [options] [--] [<namespace>]

Arguments:

# INSTALACIÓN DEL PROYECTO

Nos ubicamos en www de apache.

```
cd c:/wam64/www
```

Y dentro de esa carpeta generaremos el proyecto en una carpeta llamada miAppSymfony, esta carpeta la generará Symfony por nosotros.

En el terminal de cmd, ejecutamos:

```
composer create-project symfony/skeleton:"7.*" miAppSymfony  
cd miAppSymfony  
composer require webapp
```

# INSTALACIÓN DEL PROYECTO

La opción `-webapp` instala paquetes adicionales para brindarte todo lo que necesitas para crear una aplicación web.

Several thin, white, parallel diagonal lines are positioned in the bottom right corner of the slide, extending from the right edge towards the center.

```
c:\wamp64\www>composer create-project symfony/skeleton:"7.*" miAppSymfony
Creating a "symfony/skeleton:7.*" project at "./miAppSymfony"
Installing symfony/skeleton (v7.1.99)
  - Downloading symfony/skeleton (v7.1.99)
  - Installing symfony/skeleton (v7.1.99): Extracting archive
Created project in C:\wamp64\www\miAppSymfony
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
  - Locking symfony/flex (v2.4.7)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Installing symfony/flex (v2.4.7): Extracting archive
Generating autoload files
1 package you are using is looking for funding.
Use the `composer fund` command to find out more!

Run composer recipes at any time to see the status of your Symfony recipes.

Loading composer repositories with package information
```



Symfony operations: 17 recipes (b5bdb63b233f53b1c216dac354416239)

- Configuring symfony/webapp-pack (>=1.0): From github.com/symfony/recipes:main
- Configuring doctrine/doctrine-bundle (>=2.10): From github.com/symfony/recipes:main
- **WARNING** doctrine/doctrine-bundle (>=2.10): From github.com/symfony/recipes:main  
The recipe for this package contains some Docker configuration.

This may create/update `compose.yaml` or update `Dockerfile` (if it exists).

Do you want to include Docker configuration from recipes?

[y] Yes

[n] No

[p] Yes permanently, never ask again for this project

[x] No permanently, never ask again for this project

(defaults to y): y\_

```
c:\wamp64\www>cd miAppSymf
```

```
c:\wamp64\www\miAppSymf>composer require webapp
```

```
./composer.json has been updated
```

```
Running composer update symfony/webapp-pack
```

```
Loading composer repositories with package information
```

```
Restricting packages listed in "symfony/symfony" to "6.2.*"
```

```
Updating dependencies
```

```
Lock file operations: 99 installs, 0 updates, 0 removals
```

- Locking doctrine/cache (2.2.0)
- Locking doctrine/collections (2.2.2)
- Locking doctrine/dbal (3.9.3)
- Locking doctrine/deprecations (1.1.3)
- Locking doctrine/doctrine-bundle (2.13.0)
- Locking doctrine/doctrine-migrations-bundle (3.3.1)
- Locking doctrine/event-manager (2.0.1)
- Locking doctrine/inflector (2.0.10)
- Locking doctrine/instantiator (2.0.0)
- Locking doctrine/lexer (3.0.1)
- Locking doctrine/migrations (3.8.2)
- Locking doctrine/orm (3.1.0)

# TEST DE INSTALACIÓN.

La aplicación estará instalada en la carpeta miAppSymfony, y para probar solo hay que escribir en el navegador:

`localhost/miAppSymfony/public.`

Si todo ha ido bien debería salir una pantalla como la que se muestra a continuación.



localhost/miAppSymfony/public/



B



# Welcome to Symfony 7



You are using Symfony **7.1.8** version



Your application is ready at:

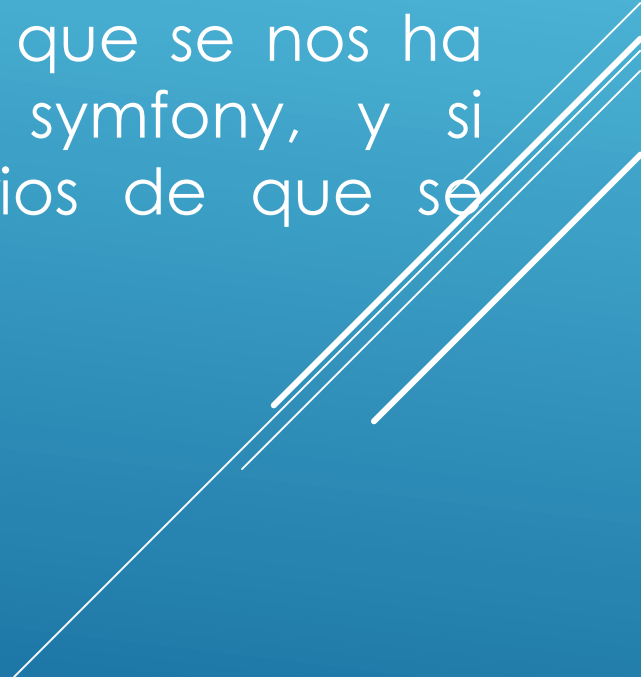
C:\wamp64\www\miAppSymfony\













You are seeing this page because the homepage URL is not configured

# CARPETA DEL PROYECTO

Si accedemos a la carpeta www de wamp, veremos que se nos ha generado una carpeta nueva, la del proyecto symfony, y si accedemos a su interior veremos todos los directorios de que se compone nuestro proyecto

Several thin, white, parallel diagonal lines are drawn across the bottom right corner of the slide, extending from the right edge towards the bottom.

equipo > Disco local (C:) > wamp64 > www

Nombre	Fecha de modifica...	Tipo	Ta
 miAppSymfony	26/11/2024 17:29	Carpeta de archivos	
 wamplanguages	03/01/2024 21:55	Carpeta de archivos	
 wampthemes	03/01/2024 21:55	Carpeta de archivos	
 add_vhost.php	10/10/2023 9:43	Archivo PHP	
 composer.phar	26/11/2024 13:29	Archivo PHAR	
 composer-setup.php	26/11/2024 13:29	Archivo PHP	
 favicon	31/12/2010 8:40	Icono	
 index.php	19/10/2023 14:43	Archivo PHP	
 test_sockets.php	21/09/2015 18:30	Archivo PHP	
 testmysql.php	17/06/2021 16:48	Archivo PHP	

Este equipo > Disco local (C:) > wamp64 > www > miAppSymfony >

Nombre	Fecha de modifica...	Tipo	Tamaño
assets	26/11/2024 17:30	Carpeta de archivos	
bin	26/11/2024 17:29	Carpeta de archivos	
config	26/11/2024 17:22	Carpeta de archivos	
migrations	26/11/2024 17:29	Carpeta de archivos	
public	26/11/2024 17:22	Carpeta de archivos	
src	26/11/2024 17:28	Carpeta de archivos	
templates	26/11/2024 17:29	Carpeta de archivos	
tests	26/11/2024 17:29	Carpeta de archivos	
translations	26/11/2024 17:29	Carpeta de archivos	
var	26/11/2024 17:22	Carpeta de archivos	
vendor	26/11/2024 17:33	Carpeta de archivos	
.env	26/11/2024 17:29	Archivo ENV	2 KB
.env.test	26/11/2024 17:29	Archivo TEST	1 KB
.gitignore	26/11/2024 17:29	Archivo GITIGNORE	1 KB
compose.override.yaml	26/11/2024 17:29	Archivo YAML	1 KB
compose.yaml	26/11/2024 17:29	Archivo YAML	1 KB
composer.json	26/11/2024 17:34	Archivo JSON	4 KB
composer.lock	26/11/2024 17:33	Archivo LOCK	356 KB
importmap.php	26/11/2024 17:30	Archivo PHP	1 KB
phpunit.xml.dist	26/11/2024 17:29	Archivo DIST	2 KB

# ESTRUCTURA DEL PROYECTO

**Public** : donde estan todos los archivos publicos como los archivos javascript, css, imágenes que queramos incorporar etc.

**Src**: Contiene los ficheros que darán la funcionalidad a nuestro proyecto:

- controladores, que es lógica interna del sistema.

- Las entidades que posteriormente se convertirán en las tablas de la BBDD

- Repositorio: Proporciona una interfaz **para acceder a los datos almacenados en la base de datos**, como búsqueda, filtrado y actualización de entidades.

- Aquí también se crearán otras carpetas como la de los formularios etc.



# ESTRUCTURA DEL PROYECTO

**Templates:** donde se muestran todos los archivos html del programas (las vistas que ofrecemos al usuario final).

**Bin:** contiene los comandos symfony, desde la raíz del proyecto si escribes el comando veras la variedad de opciones que nos ofrece:  
**php bin/console.**

**Config :** archivos de configuración del sistema

# ESTRUCTURA DEL PROYECTO

**Var:** contiene la cache del sistema

**Vendor:** contiene el código de terceros, que nos ayuda a ahorrar tiempo en codificación

**env:** Es un fichero donde tenemos la configuración básica de nuestra aplicación.(Entorno, secreto, conexión a base de datos, configurar correo electrónico....)

```
# Run "composer dump-env prod" to compile .env files for production
# https://symfony.com/doc/current/best\_practices.html#use-environment-variables

###> symfony/framework-bundle ###
##APP_ENV=dev indica desarrollo local
APP_ENV=dev
APP_SECRET=d06158e5ae23182a09245bfb7a400a8f
###< symfony/framework-bundle ###

###> doctrine/doctrine-bundle ###
# Format described at https://www.doctrine-project.org/projects/doctrine-dbal/en/latest/reference/configuration.html
# IMPORTANT: You MUST configure your server version, either here or through environment variables
#
# DATABASE_URL="sqlite:///kernel.project_dir%/var/data.db"
# DATABASE_URL="mysql://app:!ChangeMe!@127.0.0.1:3306/app?serverVersion=8.0"
# DATABASE_URL="mysql://root@127.0.0.1:3306/proyecto1"
# DATABASE_URL="mysql://app:!ChangeMe!@127.0.0.1:3306/app?serverVersion=8.0"
# DATABASE_URL="postgresql://app:!ChangeMe!@127.0.0.1:5432/app?serverVersion=12"
###< doctrine/doctrine-bundle ###

###> symfony/messenger ###
# Choose one of the transports below
# MESSENGER_TRANSPORT_DSN=amqp://guest:guest@localhost:5672/%2f/messenger
# MESSENGER_TRANSPORT_DSN=redis://localhost:6379/messages
# MESSENGER_TRANSPORT_DSN=doctrine://default?auto_setup=0
###< symfony/messenger ###

###> symfony/mailer ###
#MAILER_DSN=null://null
###< symfony/mailer ###
```