**Sistema de GEOAPP**

Manual de Instalación

**Índice**

[1. Introducción 4](#_Toc135990901)

[2. Objetivo 5](#_Toc135990902)

[3. Requerimientos 6](#_Toc135990903)

[Requerimientos de software 6](#_Toc135990904)

[Requerimientos de hardware 6](#_Toc135990905)

[4. Herramientas empleadas para el desarrollo 7](#_Toc135990906)

[5. Estructura de directorios 8](#_Toc135990907)

[6. Instalación 9](#_Toc135990908)

[Instalación de PHP 9](#_Toc135990909)

[Instalación de Laravel 12](#_Toc135990910)

[Instalación de git 14](#_Toc135990911)

[Configuración Local 22](#_Toc135990912)

[7. Base de datos 27](#_Toc135990913)

[Diagrama entidad relación 27](#_Toc135990914)

[Diccionario de datos 28](#_Toc135990915)

[8. Diagrama de Flujo de Datos 32](#_Toc135990916)

[9. How to 33](#_Toc135990917)

[10. Sitios de Interés 34](#_Toc135990918)

**Figuras**

[Figura 1 Sitio web de descarga de XAMPP 9](#_Toc136071623)

[Figura 2 Selección de folder de instalación 9](#_Toc136071624)

[Figura 3 Proceso de Instalación Completado 11](#_Toc136071625)

[Figura 4 XAMPP en ejecución 11](#_Toc136071626)

[Figura 5 Sitio web de descarga de Composer 12](#_Toc136071627)

[Figura 6 Proceso de instalación Composer ,gestor de paquetes PHP 12](#_Toc136071628)

[Figura 7 Sitio Web de descarga de git 14](#_Toc136071629)

[Figura 8 Proceso de Instalación de git en Windows 14](#_Toc136071630)

[Figura 9 Directorio de Instalación 15](#_Toc136071631)

[Figura 10 Componentes de Git 15](#_Toc136071632)

[Figura 11 Directorio de acceso directo 16](#_Toc136071633)

[Figura 12 Editor por defecto Git 16](#_Toc136071634)

[Figura 13 Ramas por defecto de repositorio 17](#_Toc136071635)

[Figura 14 Git form the command line and also from 3rd-pary software 17](#_Toc136071636)

[Figura 15 conexiones SSL/TLS recomendadas por defecto 18](#_Toc136071637)

[Figura 16 Fin de Línea en git 18](#_Toc136071638)

[Figura 17 Emulador de git bash 19](#_Toc136071639)

[Figura 18 Comportamiento de git pull 19](#_Toc136071640)

[Figura 19 Gestión de credenciales de git 20](#_Toc136071641)

[Figura 20 Opciones Adicionales de git 20](#_Toc136071642)

[Figura 21 Opciones de configuraciones experimental 21](#_Toc136071643)

[Figura 22 Pantalla Fin de instalación 21](#_Toc136071644)

[Figura 23 Servidor de repositorio gitlab 22](#_Toc136071645)

[Figura 24 Opciones de clonación de repositorio 22](#_Toc136071646)

[Figura 25 Sitio Web de descarga de Node 23](#_Toc136071647)

[Figura 26 Instalador de Node en carpeta de descarga 24](#_Toc136071648)

[Figura 27 Ventana de Instalación de Node 24](#_Toc136071649)

[Figura 28 Proceso de Instalación de Node 25](#_Toc136071650)

[Figura 29 Archivo de configuración .env 26](#_Toc136071651)

# Introducción

El presente manual técnico proporciona una guía detallada sobre la instalación y configuración de los componentes necesarios para el correcto funcionamiento del aplicativo web de administración de laboratorio geotecnico, basado en el framework Laravel. Este sistema está diseñado para facilitar la carga eficiente y precisa de datos desde archivos Excel, brindando una solución ágil y confiable para la gestión de información.

El manual abarca tanto los requisitos de hardware como los softwares necesarios para la instalación, asegurando que el entorno esté correctamente configurado. Se describen los pasos necesarios para la preparación del entorno de desarrollo, incluyendo la instalación de Laravel, la configuración del servidor web y la configuración de la base de datos compatible.

Además, se proporcionan instrucciones detalladas sobre la instalación y configuración de las dependencias y bibliotecas requeridas para el manejo de archivos Excel, así como las herramientas adicionales que se recomiendan para maximizar el rendimiento y la seguridad del sistema.

Al seguir este manual técnico de instalación, los usuarios podrán configurar rápidamente el entorno necesario y utilizar el sistema de administración de laboratorio geotécnico desarrollado con Laravel de manera eficiente y confiable, maximizando así la productividad y la calidad de los procesos de carga de información

# Objetivo

El objetivo del manual técnico es proporcionar una referencia clara y detallada de los componentes del sistema de importación de datos con Laravel, incluyendo hardware, software, bibliotecas, framework y otros elementos relevantes. Esto permitirá a los responsables del mantenimiento identificar rápidamente los componentes utilizados y comprender su funcionalidad, facilitando así el mantenimiento efectivo del sistema.

# Requerimientos

## Requerimientos de software

Para la Implementación de este sistema el servidor debe cumplir con los siguientes softwares previamente configurados:

* PHP Versión 8.1.12
* Laravel 9
* Servidor Apache v2.4.54
* Servidor de base de datos SQL server 2008
* Composer
* NPM
* Configuración de DLL para SQL Server
* Configurar característica de SOAP
* Sistema operativo Windows 7 o Linux

## Requerimientos de hardware

* Procesador: 2.8 GHz Intel Core I5.
* Memoria RAM 8 GB
* Disco duro 500GB
* 20 GB de espacio en disco en el servidor web
* Una conexión a Internet estable es necesaria para descargar las dependencias del proyecto, así como para interactuar con servicios externos

# Herramientas empleadas para el desarrollo

**PHP**

PHP es un [lenguaje de programación](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/que-son-los-lenguajes-de-scripting/) de código abierto del lado del servidor que se utiliza principalmente para **crear páginas web dinámicas**.

PHP 8.0 es una actualización importante del lenguaje PHP que contiene nuevos recursos y optimizaciones incluyendo argumentos nombrados, tipos de uniones, atributos, promoción de propiedades constructivas, expresiones match, operador nullsafe, JIT también mejoras en el sistema de tipos, manejo de errores y consistencia en general.

**COMPOSER**

Es un gestor de librerías enfocado a PHP encargado de instalar paquetes de los que dependen los proyectos, este administrará y actualizará automáticamente en un directorio en específico.

**LARAVEL**

Laravel es un framework enfocado a aplicaciones web. Este presenta una estructura para el inicio de la creación de aplicaciones. Posee una inyección de dependencias, una capa de abstracción a la base de datos y pruebas de integración. Facilitando una mejor experiencia de desarrollo y seguridad en la información.

**APACHE**

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual según la normativa RFC 2616.

**BOOTSTRAP 4**

Bootstrap es un framework front-end utilizado para desarrollar aplicaciones web y sitios Mobile first, El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

# Estructura de directorios

La estructura de directorios de Laravel sigue una convención bien establecida que organiza los archivos de manera lógica y modular. A continuación, se muestra la estructura de directorios básica de Laravel:

* **Carpeta raíz (\):** Contiene archivos y directorios esenciales para el funcionamiento del framework.
* **\app:** Contiene los archivos principales de la aplicación, como modelos, controladores, proveedores de servicios y otros componentes personalizados.
* **\bootstrap:** Contiene archivos de inicio y configuración temprana de la aplicación. Incluye el archivo app.php, que carga la configuración inicial de Laravel.
* **\config:** La carpeta config contiene archivos de configuración de la aplicación, donde puedes especificar opciones como la conexión de la base de datos, la configuración del correo electrónico, servicios externos, etc.
* **\database:** Contiene archivos relacionados con la base de datos, como migraciones, semillas (seeds) y factories (fábricas).
* **\public:** La carpeta public es el punto de entrada de tu aplicación. Contiene el archivo index.php que redirige todas las solicitudes a tu aplicación.
* **\resources:** Contiene los recursos de tu aplicación, como vistas, archivos de lenguaje, archivos de estilo CSS y archivos de JavaScript.
* **\routes:** Contiene archivos de rutas de la aplicación, donde están definidas las rutas y su correspondiente lógica de controlador.
* **\storage:** Se utiliza para almacenar archivos generados por la aplicación, como archivos de registro, archivos cargados por los usuarios y cachés.
* **\tests:** La carpeta tests contiene archivos de pruebas.
* **\vendor:** La carpeta vendor contiene las dependencias del proyecto que son administradas por Composer.

# Instalación

## Instalación de PHP

* Descargar XAMPP del sitio oficial , <https://www.apachefriends.org/download.html>.
* Seleccionar la Versión de PHP 8.02.

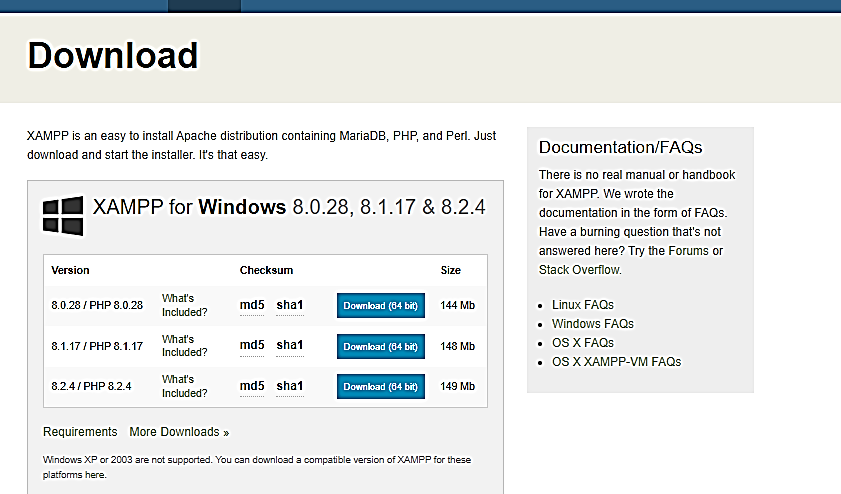


Figura 1 Sitio web de descarga de XAMPP

* Ejecuta el instalador descargado y sigue las instrucciones del asistente de instalación y configurar XAMPP

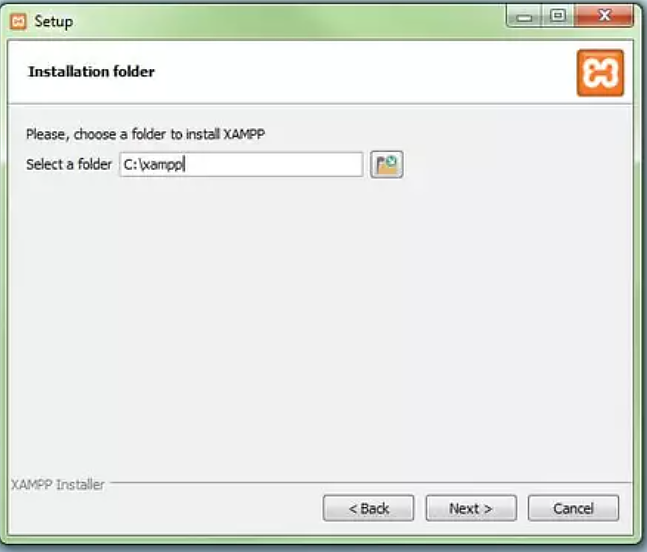
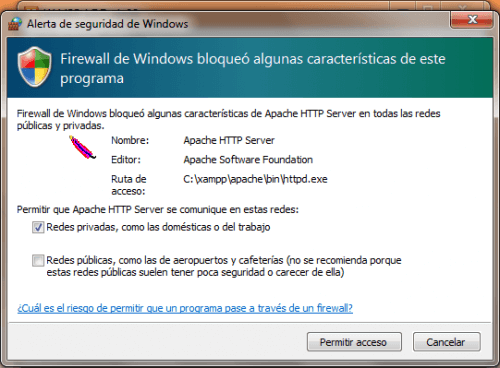


Figura 2 Selección de folder de instalación





* Finalizar instalación, Una vez extraídos e instalados todos los componentes, cerrar el asistente con “**Finish**”.

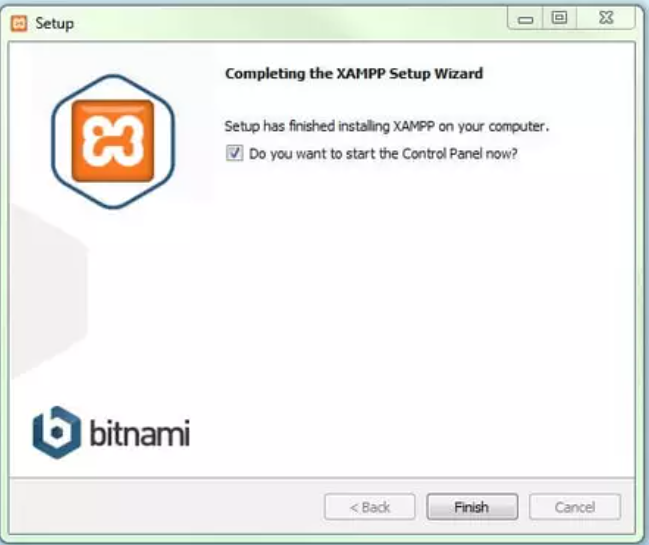


Figura 3 Proceso de Instalación Completado



Figura XAMPP en ejecución

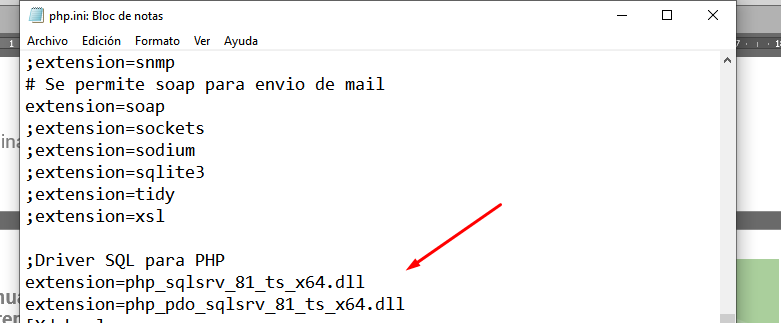
* Configuración de PHP para SQL server

Se requiere los dos archivos de librerías, dependiendo de la versión de tu PHP.

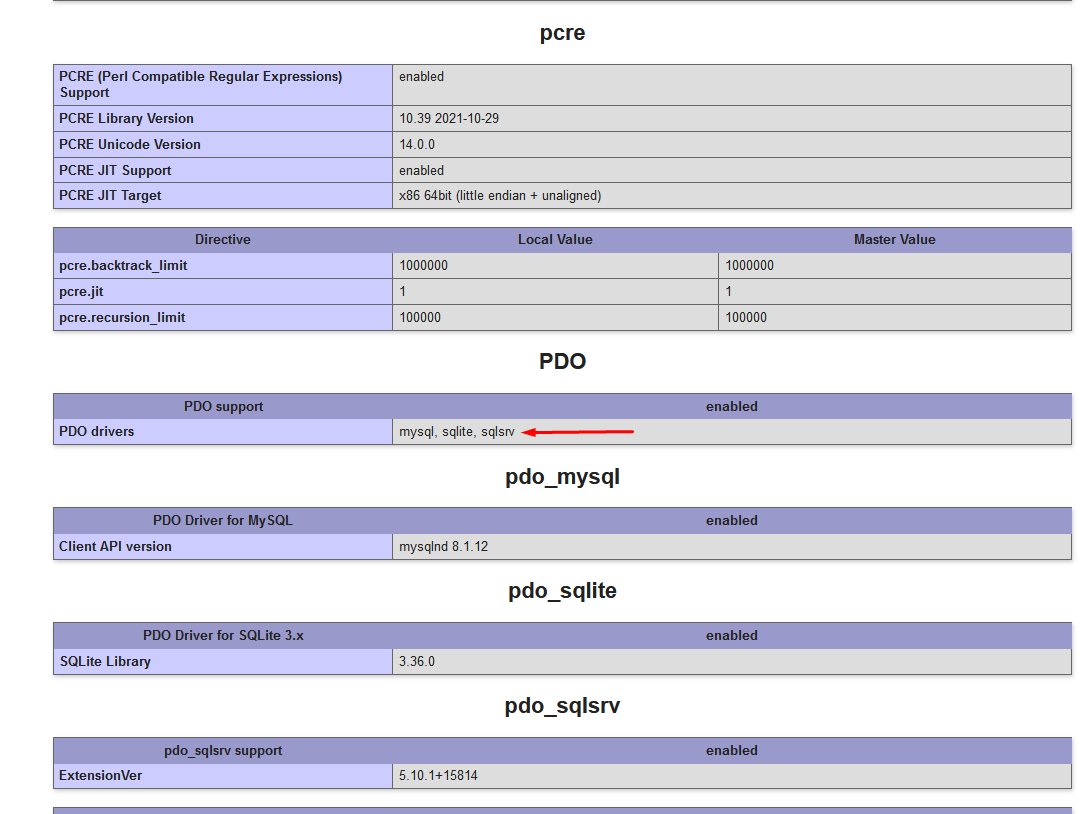
Instalar la extensión de SQL Server en PHP.



Editar el archivo «php.ini»:



Reiniciar Apache y verificar que la extensión haya sido reconocida desde PHPINFO ():



## Instalación de Composer

* Descarga Composer desde <https://getcomposer.org/download/>.
* Ejecuta el instalador descargado y sigue las instrucciones del asistente de instalación.

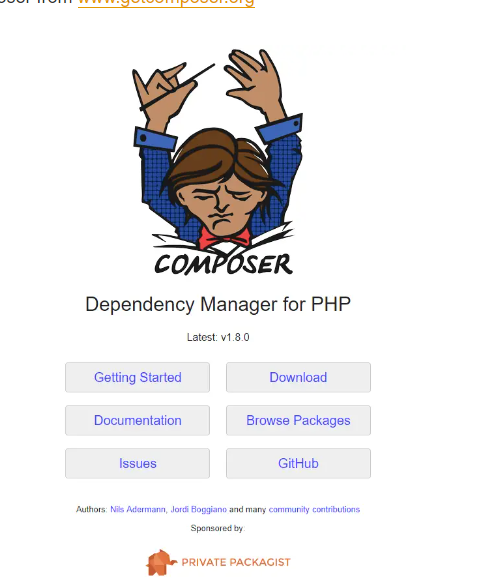


Figura 5 Sitio web de descarga de Composer

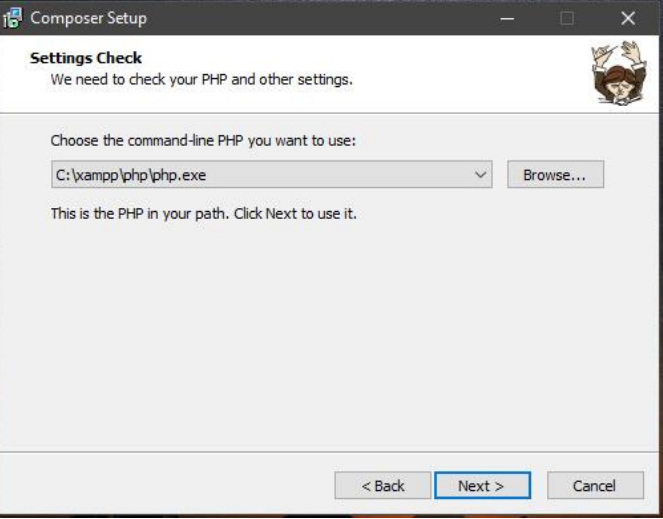
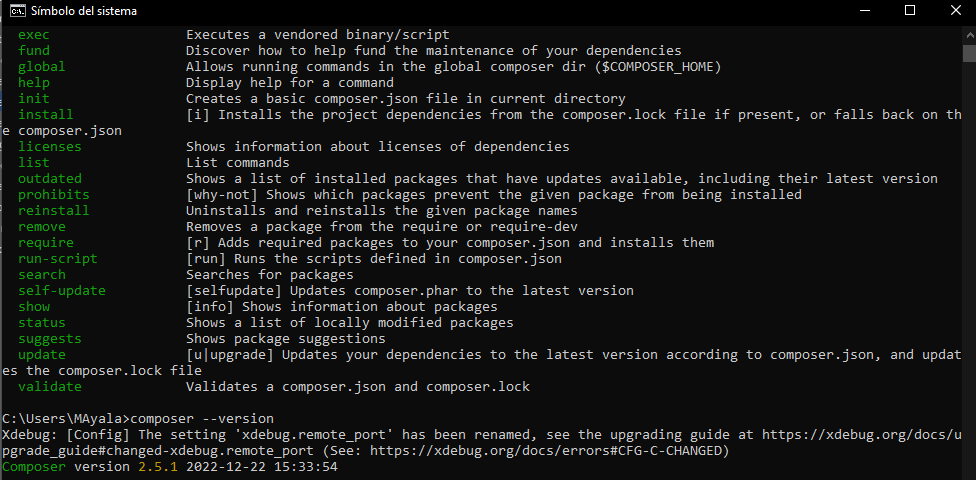


Figura 6 Proceso de instalación Composer ,gestor de paquetes PHP

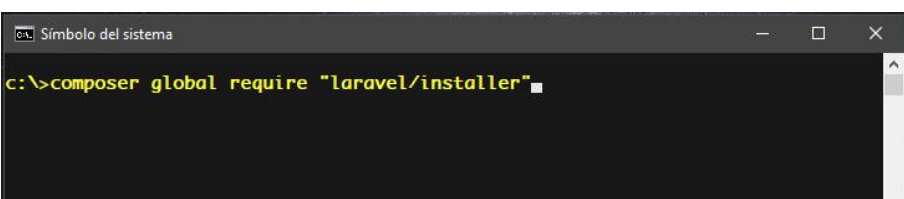
* Verifica la instalación de Composer abriendo una nueva ventana de línea de comandos (CMD) y ejecutando el siguiente comando.

composer --version



## Instalación de Laravel

* Abre una ventana de línea de comandos (CMD).
* Ahora se debe ingresar el comando: **composer global require “laravel/installer”,** para instalar el Laravel de manera global.



* Verifica la instalación de Laravel ejecutando el siguiente comando

laravel --version

## Instalación de git

Para mantener el código actualizado debemos tener configurado git

* Descargar del sitio web : <https://git-scm.com/downloads> .

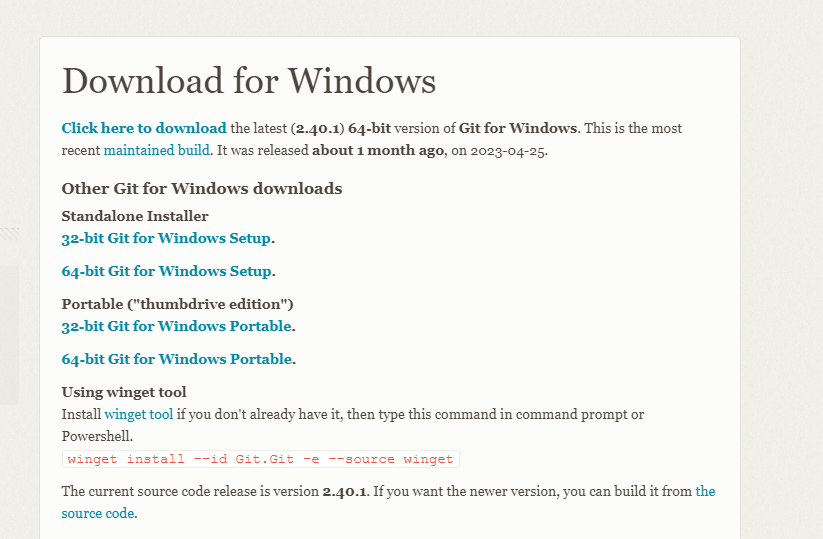


Figura 7 Sitio Web de descarga de git

* Aceptar los términos y las condiciones que se muestran en pantalla y haz clic en **Next** para continuar.

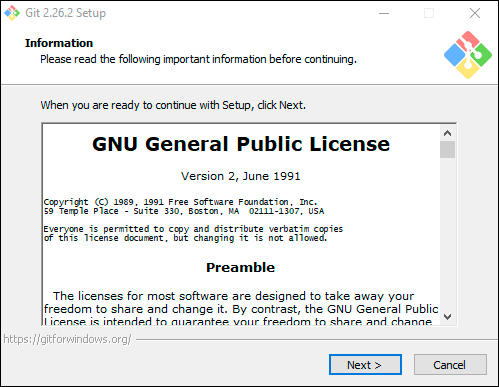


Figura 8 Proceso de Instalación de git en Windows

* Luego selecciona el directorio de instalación y haz clic en **Next** para continuar.

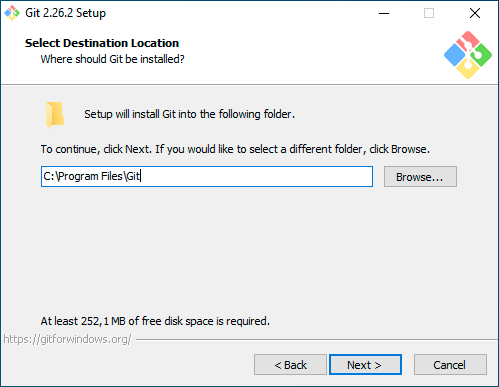
****

Figura 9 Directorio de Instalación

* Seleccionar los componentes que se instalarán. Es recomendable que selecciones los componentes que integrarán Git con el explorador de Windows.

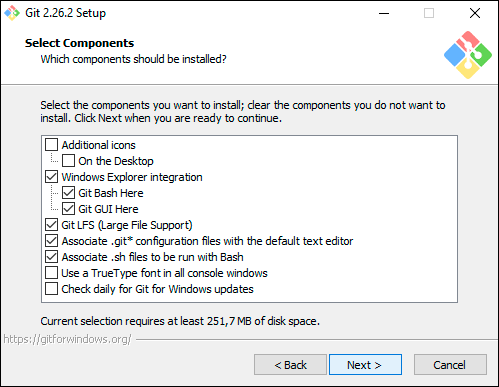
****

Figura 10 Componentes de Git

* Seleccionar el directorio en el que se crearán los accesos directos del menú de inicio Haz clic en Next para continuar.

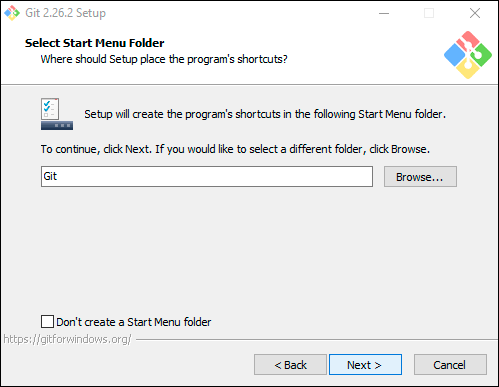
**[](https://www.neoguias.com/wp-content/uploads/2020/05/instalar-git-windows-04.png)**

Figura 11 Directorio de acceso directo

* Seleccionar el editor que usará Git de forma predeterminada. Haz clic en **Next** para continuar.

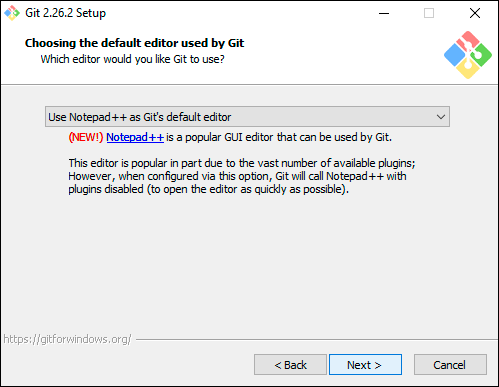
[](https://www.neoguias.com/wp-content/uploads/2020/05/instalar-git-windows-05.png)

Figura 12 Editor por defecto Git

* Reemplazar la opción «Let Git decide» y haz clic en Next para continuar.

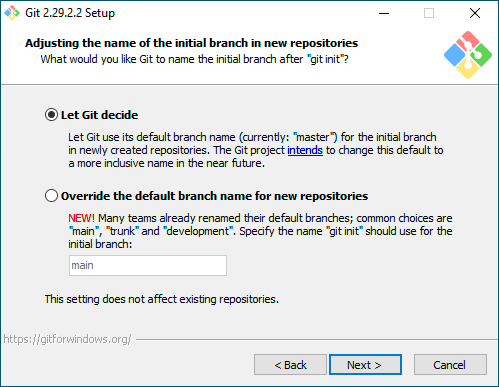


Figura 13 Ramas por defecto de repositorio

* Seleccionar las opciones de configuración «Git form the command line and also from 3rd-pary software», Haz clic en Next para continuar.

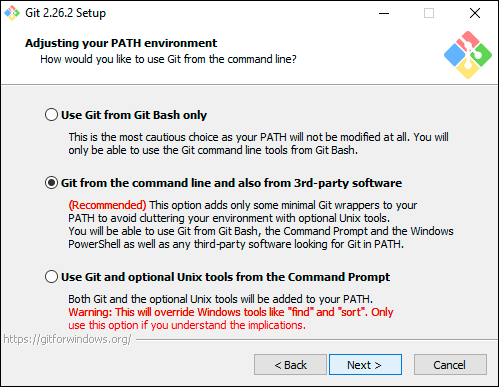
**[](https://www.neoguias.com/wp-content/uploads/2020/05/instalar-git-windows-06.png)**

Figura 14 Git form the command line and also from 3rd-pary software

* Seleccionar el tipo de librería que usará Git para gestionar las conexiones SSL/TLS.

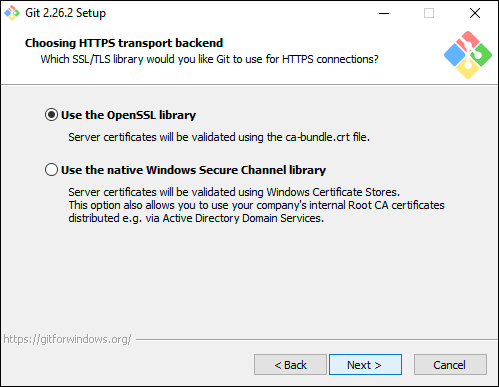
****

Figura 15 conexiones SSL/TLS recomendadas por defecto

* Seleccionar el trato que Git dará a los marcadores de final de línea. Haz clic en Next para continuar.

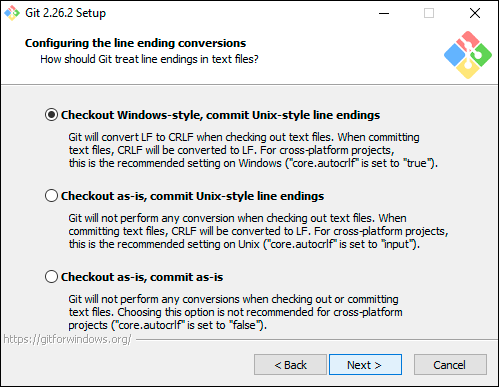
****

Figura 16 Fin de Línea en git

* Seleccionar interfaz MinTTY, que es la opción por defecto y la recomendada, Haz clic en Next

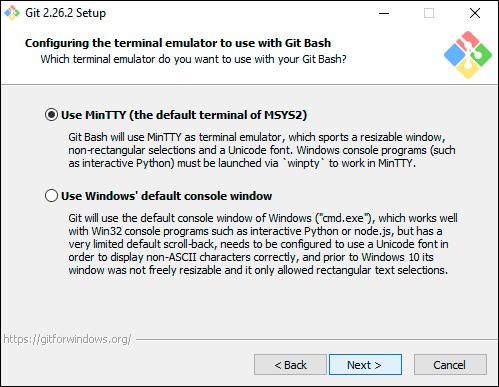
****

Figura 17 Emulador de git bash

* Luego se te preguntará por el funcionamiento que tendrá el comando git pull. Escoger la opción por defecto.

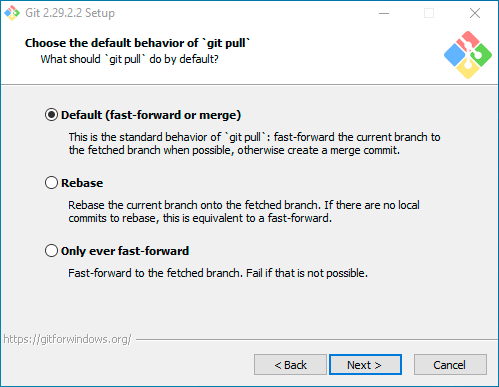


Figura 18 Comportamiento de git pull

* Reemplazar la aplicación que se usará para gestionar las credenciales de Git. Puedes Reemplazar entre Git Credential Manager Core

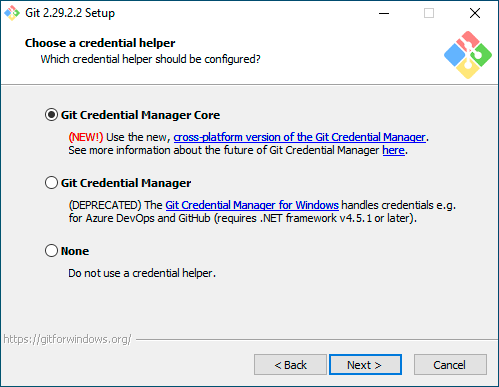


Figura 19 Gestión de credenciales de git

* Selección de caché de Git y activa también la compatibilidad con enlaces simbólicos. Luego haz clic en Next para continuar.

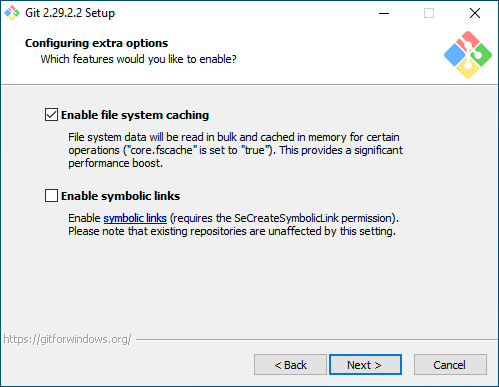


Figura 20 Opciones Adicionales de git

* Desactiva esta opción experimental finalmente haz clic en **Install**.

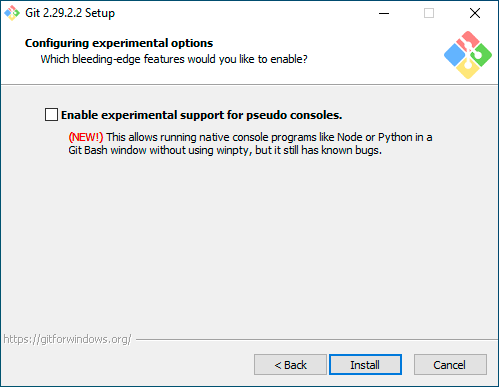


Figura 21 Opciones de configuraciones experimental

* Para finalizar la instalación, haz clic en **Next**

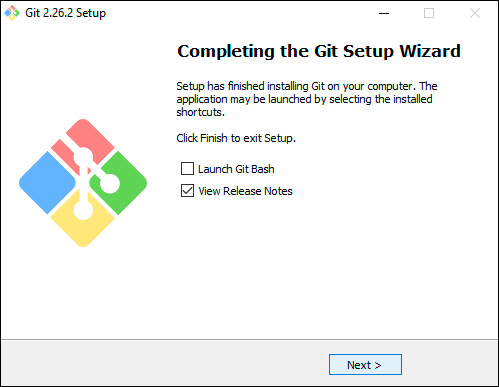
****

Figura 22 Pantalla Fin de instalación

## Configuración Local

* Abrir terminal y clonar repositorio en directorio **C:\xampp\htdocs**

Git clone http://gitlab.credifacilnicaragua.com/desarrollo/importadata

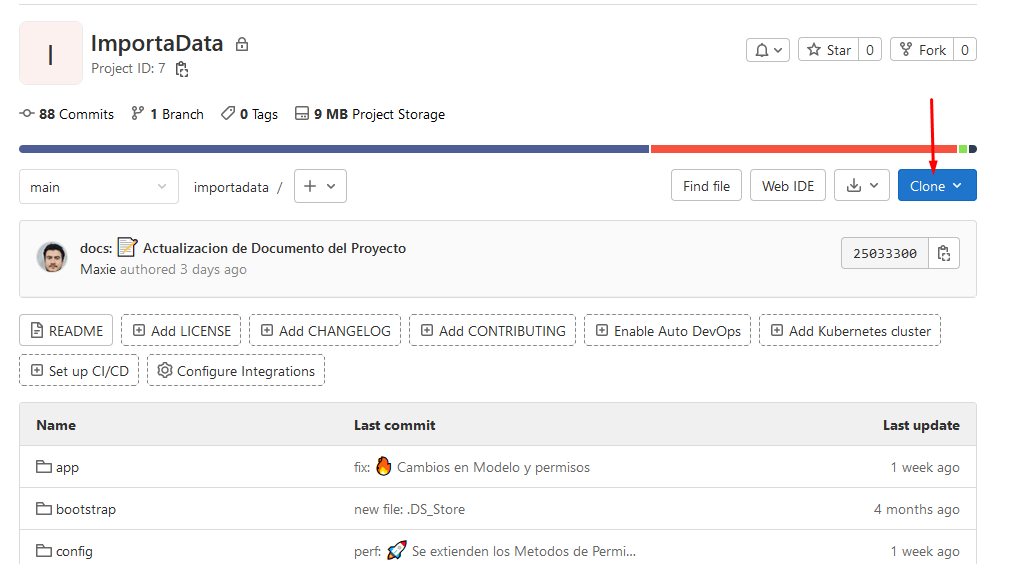


Figura 23 Servidor de repositorio gitlab

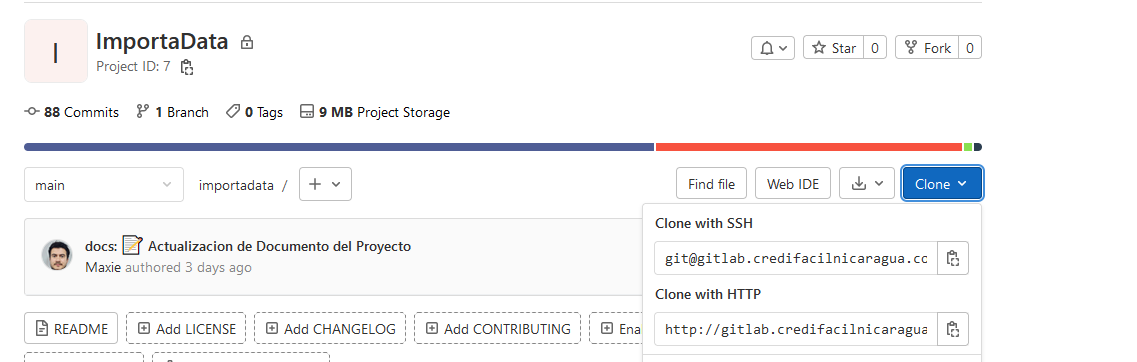
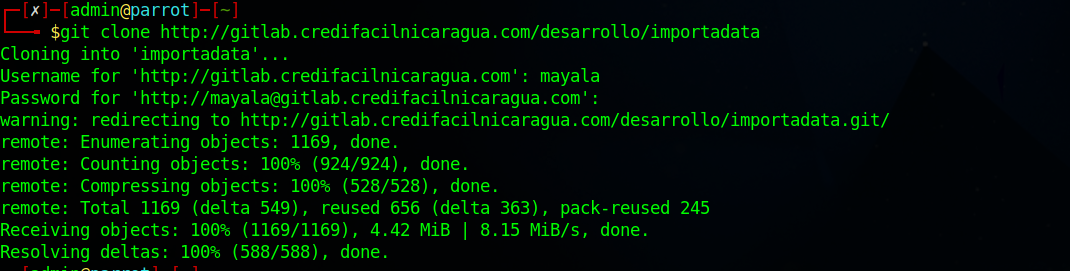
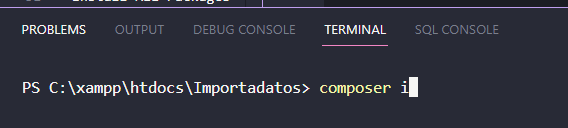
****

Figura 24 Opciones de clonación de repositorio

* En terminar git solicitara las credenciales, ingresar usuario y contraseña



* Instalar todas las dependencias de Laravel



* Instalar Node versión 18.16.0.

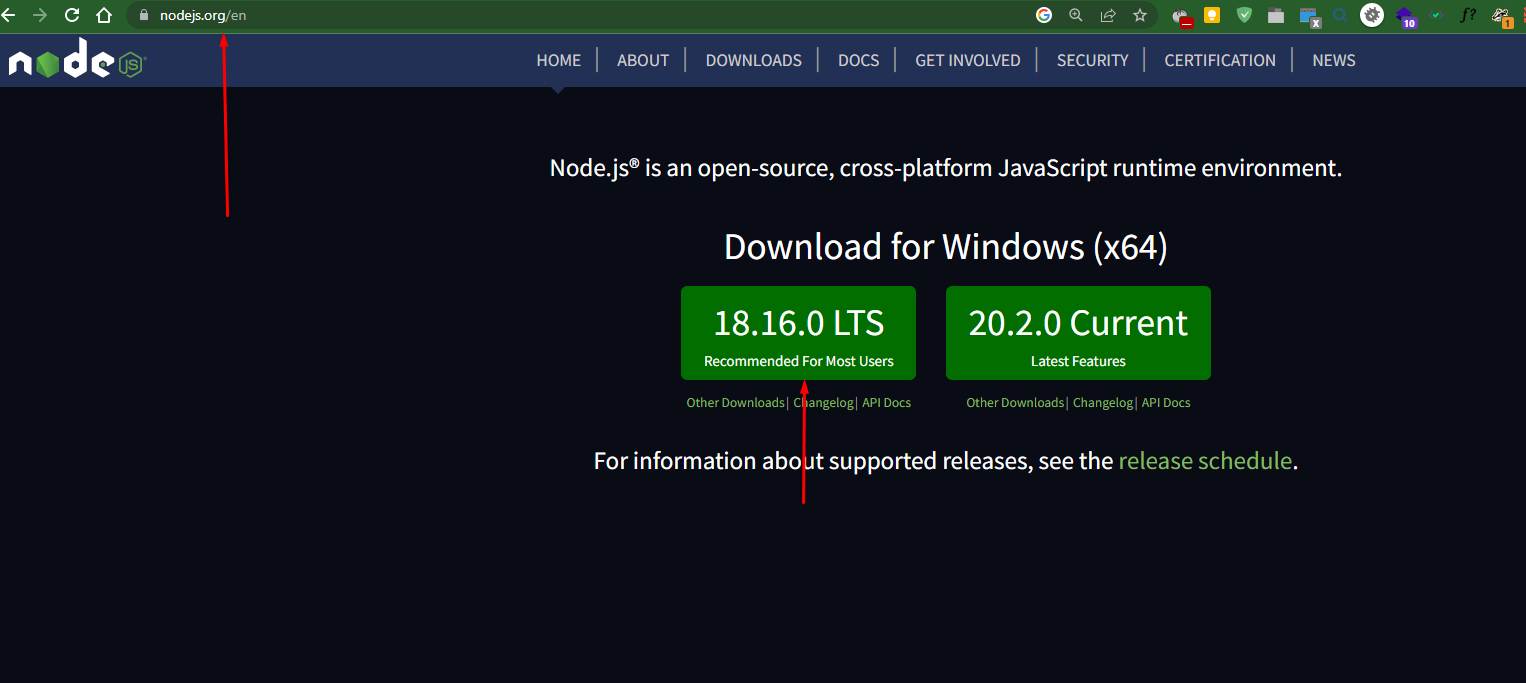


Figura 25 Sitio Web de descarga de Node

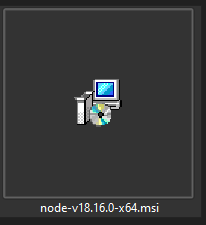


Figura 26 Instalador de Node en carpeta de descarga

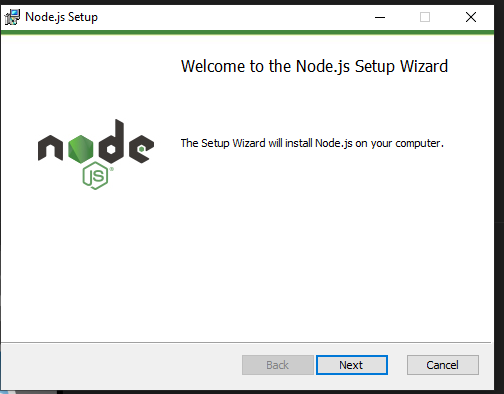


Figura 27 Ventana de Instalación de Node

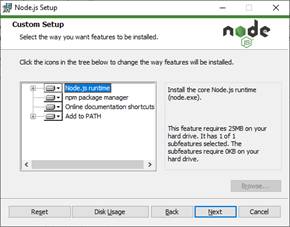
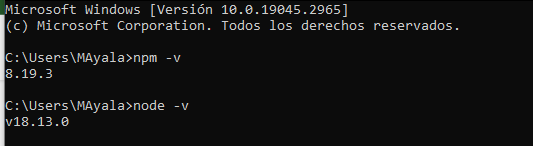
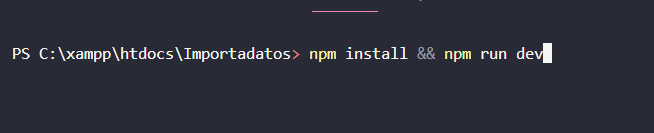


Figura 28 Proceso de Instalación de Node

* Comprobar Versión de **NPM.**



* Instalar dependencias **NPM** del proyecto.



* Crear archivo “.env” que contendrá las variables de entorno, ubicado en la raíz del proyecto.

cp .env.example .env

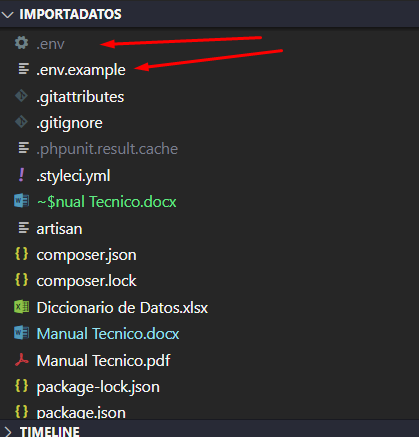
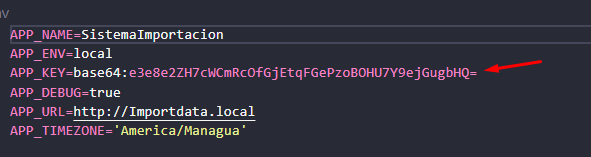


Figura 29 Archivo de configuración .env

* Generar llave de identificación de aplicación.

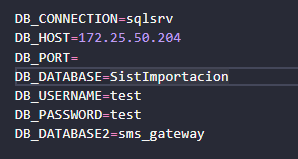
php artisan key:generate



* Actualizar el archivo “.env” con las credenciales de acceso a base de datos.

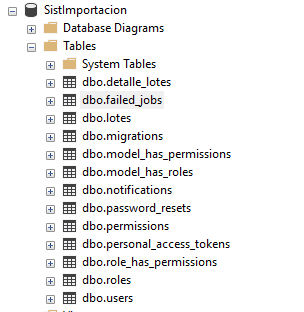
|  |  |
| --- | --- |
| Variable | Descripción |
| DB\_CONNECTION | Tipo de servidor base de datos (en nuestro caso el tipo es : sqlsrv) |
| DB\_HOST | Servidor de base de datos |
| DB\_PORT | En esta línea se debe colocar el puerto de servidor de base de datos. |
| **DB\_DATABASE** | Colocar el nombre de la base de datos principal SistImportacion |
| **DB\_DATABASE2** | Colocar el nombre de la base de datos de sms Gateway |
| **DB\_USERNAME** | Nombre de usuario de tu Base de Datos |
| **DB\_PASSWORD** | Contraseña de tu Base de Datos. |

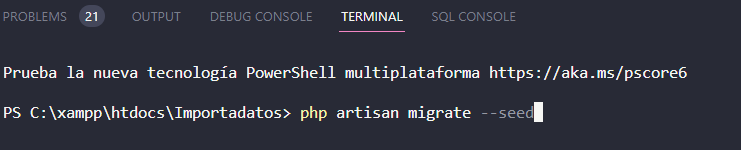
* servidor de base de datos, bases de datos, username, password.

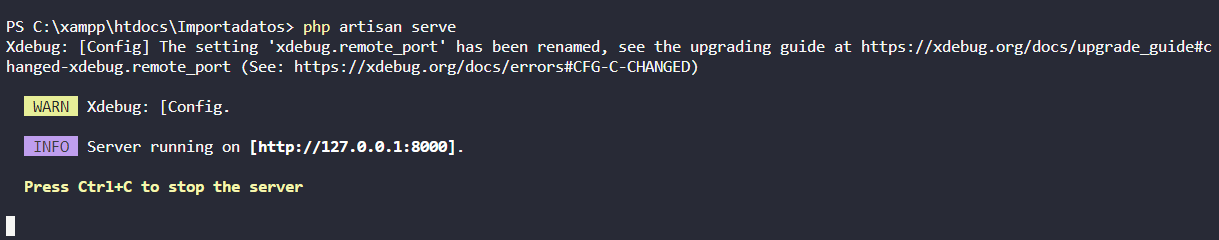


* Acceder a servidor de base de datos, crear la base de datos de la aplicación.

Nombre de base de datos: **SistImportacion**



* En consola de Windows y ejecutar el comando php artisan migrate, el cual se encargará de ejecutar los archivos de migración de laravel que crean todas las tablas necesarias para la base de datos. 
* Cuando terminamos nuestra configuración del proyecto el último paso es ejecutar el comando “php artisan serve”, donde nos proporciona una url local para que nuestro proyecto funcione en el navegador web, donde nos mostrara la vista de nuestro login, de esta manera comprobamos que nuestro proyecto está ejecutándose con todas las dependencias y bases de datos



* Accedemos al enlace y muestra la página de autenticación de la aplicación.

Ingresar con usuario administrador:

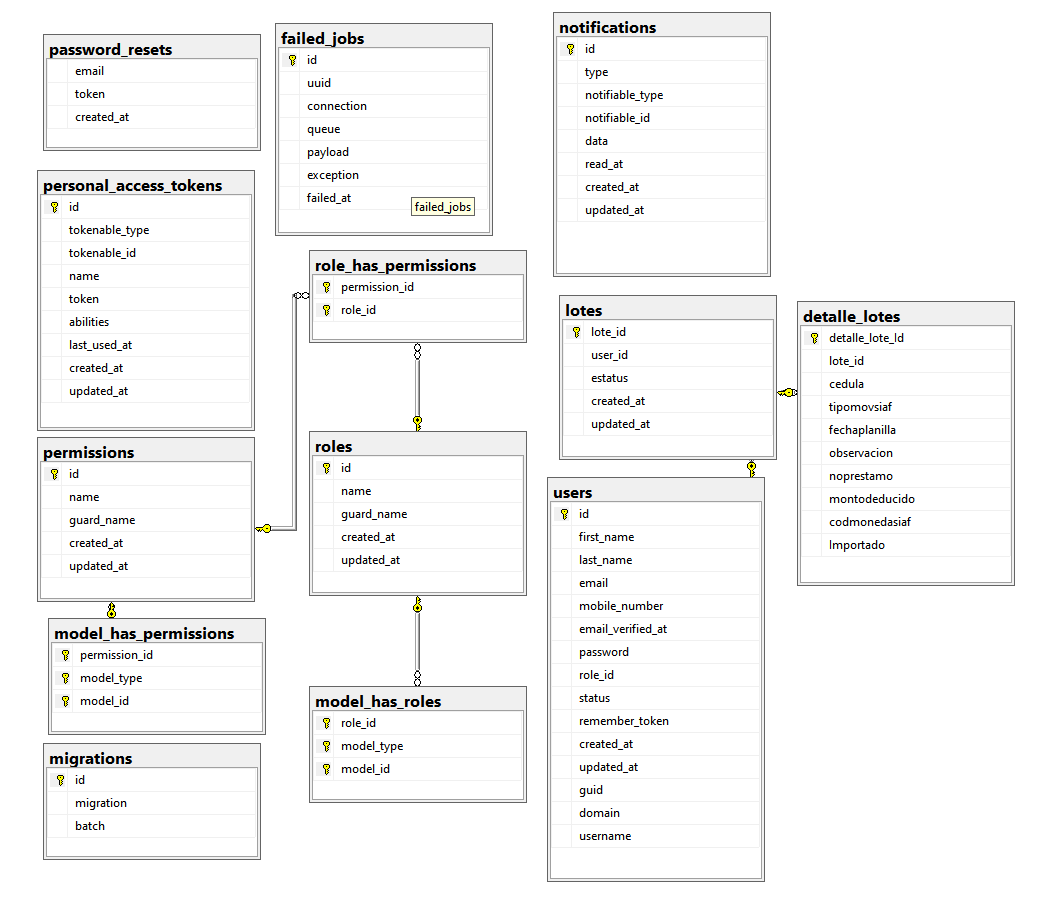
Username - admin

Password - Admin@123#



# Base de datos

## Diagrama entidad relación



## Diccionario de datos

detalle\_lotes



Lotes



users



Roles



Notifications



|  |  |  |  |
| --- | --- | --- | --- |
| Permissions  **permissions** | | | |
| **Nombre** | **Tipo de dato** | **Nulo** | **Comentario** |
| id | bigint | N |  |
| name | nvarchar(255) | N |  |
| guard\_name | nvarchar(255) | N |  |
| created\_at | datetime | Y |  |
| updated\_at | datetime | Y |  |

failed\_jobs



Migrations



role\_has\_permissions



model\_has\_permissions



model\_has\_roles



password\_resets



personal\_access\_tokens



# Diagrama de Flujo de Datos

# How to

# Sitios de Interés

How to Deploy Laravel Projects to Live Server: The Ultimate Guide

<https://laraveldaily.com/post/how-to-deploy-laravel-projects-to-live-server-the-ultimate-guide>

Guía como Montar Servidor Laravel en Windows Server 2019

<https://www.youtube.com/watch?v=YMMFfAad3y8>

Git en el Servidor - Configurando Git en un servidor

<https://git-scm.com/book/es/v2/Git-en-el-Servidor-Configurando-Git-en-un-servidor>

Laravel 8 Admin Panel Starter Kit | Tech-Admin | Laravel 8 + Bootstrap 4 | Laravel 8 Admin Template

<https://www.youtube.com/watch?v=AQeWj89c6FE&t=166s>