

Paso 1: Instalación y Configuración de MySQL

1. Instalación de MySQL en WSL:

Primero, instalamos MySQL usando los siguientes comandos:

```
sudo apt update  
sudo apt install mysql-server
```

Verificamos que MySQL esté corriendo:

```
sudo systemctl status mysql
```

Si no está corriendo, iniciamos el servicio con:

```
sudo systemctl start mysql
```

Paso 2: Creación de la Base de Datos y Usuario

2. Creación de la Base de Datos y Usuario:

Accedemos a MySQL como usuario root:

```
CREATE DATABASE example_app;
```

Dentro de MySQL, creamos una nueva base de datos:

```
CREATE USER 'example_user'@'localhost' IDENTIFIED BY 'secure_password';  
GRANT ALL PRIVILEGES ON example_app.* TO 'example_user'@'localhost';  
FLUSH PRIVILEGES;  
EXIT;
```

Paso 3: Configuración en Laravel

3. Configurar Laravel para Usar la Base de Datos:

Editamos el archivo .env para configurar la conexión a la base de datos:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=example_app
DB_USERNAME=example_user
DB_PASSWORD=secure_password
```

Paso 4: Ejecutar las Migraciones y Problemas Encontrados

4. Ejecución de las Migraciones y Problemas Encontrados:

Navegamos a la carpeta del proyecto Laravel:

```
cd /ruta/a/tu/proyecto/example-app
```

Ejecutamos las migraciones con:

```
php artisan migrate
```

Problema: Advertencias sobre la extensión pdo_mysql:

```
PHP Warning: Unable to load dynamic library 'pdo_mysql' ...
```

Solución: Instalar pdo_mysql y deshabilitar pdo_sqlsrv si no es necesario:

```
sudo apt-get install php-mysql
sudo nano /etc/php/8.2/cli/php.ini
# Comentar la línea que carga `pdo_sqlsrv.so`
;extension=pdo_sqlsrv.so
```

Verificamos extensiones cargadas con:

```
php -m
```

Resultado: Las migraciones se ejecutan correctamente tras resolver los problemas.