

**PLAN DE MIGRACIÓN DE BASES DE DATOS  
PROYECTO: COTTON & CO SWEATERS**

**INTEGRANTES:**

**ANA MARIA RIAÑO CARO  
ANIBAL YESITH OVIEDO MADERA  
ALEXIS LUQUE OROZCO  
CARLOS DANIEL GIRALDO NARANJO  
DAVID SANTIAGO VARGAS OYOLA  
KAROL VALENTINA ÁVILA QUINTERO**

**INSTRUCTOR:**

**JAVIER LEONARDO PINEDA URIBE**

**BOGOTÁ 30 DE Agosto DE 2022  
SERVICIO NACIONAL DE APRENDIZAJE – SENA  
CENTRO ELECTRICIDAD, ELECTRÓNICA Y TELECOMUNICACIONES – CEET  
ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN**

## **Tabla de Contenido**

INTRODUCCIÓN	1
JUSTIFICACIÓN	2
ANÁLISIS PREVIO DE LA BASE DE DATOS ACTUAL.	3
Número de Objetos de conversión.	3
Número de Registros filas:	3
Número de Atributos (NA)	3
Número de claves foráneas. Referential Degree. (RD)	3
ANÁLISIS DE MIGRACIÓN	4
ESTRATEGIA DE MIGRACIÓN (PLAN DE MIGRACIÓN).	5
ELABORAR PLAN DE EJECUCIÓN	5
Plan de Marcha atrás.	5
Mecanismo de escalado y manejo de problemas.	5
Plan de Implantación.	6
BIBLIOGRAFÍA	6

## **Tabla de Ilustraciones.**

Ilustración 1 Tipos de datos.

4

## **INTRODUCCIÓN**

La migración de BD es un proceso que se lleva a cabo para trasladar datos almacenados previamente de un gestor de base de datos a otro o desde el mismo gestor pero a versiones diferentes, por ello este documento se verá reflejado los pasos que se van a tomar para migrar la base de datos desde MySQL a PostgreSQL, a través del programa FullConvert. Además, se tomará en cuenta los otros tipos de migración de bases de datos, de una ubicación a otra pero con el mismo motor de base de datos.

Las copias de seguridad son de gran importancia debido a que el activo más importante es la información. Hoy en día todo depende de la información guardada en los servidores, pero si por alguna razón, por ejemplo desastres naturales, robos, incendios, inundaciones, fallos informáticos, o hay una pérdida de ésta información, sería casi imposible volver a recuperarla de no tener copias de seguridad.

## **JUSTIFICACIÓN**

La información es muy esencial en cualquier sistema por lo cual es una buena opción que este sea más versátil a la hora de transportarla a otra ubicación ya sea por la existencia de Big Data que se deba introducir esta información en un motor con una mejor opción para el manejo de esta extensa información.

## **ANÁLISIS PREVIO DE LA BASE DE DATOS ACTUAL.**

Antes de realizar la copia de seguridad, se debe realizar un análisis previo del sistema gestor de base de datos de MySQL que es el actual para así verificar posteriormente con la copia de seguridad o con la migración al sistema gestor de base de datos final PostgreSQL. Esto se realiza para tener una visión general de las ventajas y/o desventajas de realizar dicha migración, además se estima el nivel de complejidad de llevar a cabo dicho proceso. Estos elementos son:

### **Número de Objetos de conversión.**

En la base de datos MySQL hay objetos que no tienen correspondencia directa con respecto a la base de datos PostgreSQL por tanto dicho proceso se vuelve un poco complejo. Es por eso que es tan importante conocer qué tan complejo es dicha conversión y así tomar medidas para prever el impacto de la migración.

Los objetos de la base de datos cuya conversión a lenguaje PostgreSQL, de nivel de complejidad bajo son: tablas, índices, secuencias, vistas.

### **Número de Registros filas:**

La cantidad de registros en una base de datos es importante en este análisis previo, puesto que no es lo mismo realizar una migración de cien registros a más de mil millones de registros. Tener en cuenta que el número de registro de filas afecta tiempos de actualización, Back up, disparadores, etc.

- **Administrador:**

```
select count(*) from administrador;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
5

- **Ciudad:**

```
select count(*) from ciudad;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
15

- **Estado:**

```
select count(*) from estado;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
2

- **Factura:**

```
select count(*) from factura;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
0

- **Producto:**

```
select count(*) from producto;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
0

- **Usuario:**

```
select count(*) from usuario;
```

☐ Perfilando [ [Editar en línea](#) ] [ [Editar](#) ] [ [Explicar SQL](#) ] [ [Crear código PHP](#) ] [ [Actualizar](#) ]

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

count(*)
2



## Número de Atributos (NA)

Otro de los indicadores de complejidad se calcula en base a la cantidad de campos de las tablas. Se recomienda ejecutar consultas que permitan ver el promedio en general de las columnas, la cantidad máxima y mínima de columnas por tabla. No es lo mismo tener una tabla con 200 campos a una tabla con 2 campos, el manejo en cuanto a índices, restricciones, es diferente.

- **Administrador:**

```
describe administrador;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
idAdministrador	varchar(10)	NO	PRI	NULL	
docType	varchar(30)	NO	PRI	NULL	
firstName	varchar(30)	YES		NULL	
secondName	varchar(30)	YES		NULL	
surname	varchar(30)	YES		NULL	
secondSurname	varchar(30)	YES		NULL	
indicativo	varchar(4)	YES		NULL	
phone	varchar(15)	YES		NULL	
correo	varchar(30)	YES		NULL	
direccion	varchar(60)	YES		NULL	
pass	varchar(60)	YES		NULL	
idCiudad	varchar(3)	YES	MUL	NULL	
idEstado	bit(1)	YES	MUL	NULL	

- **Ciudad:**

```
describe ciudad;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
idCiudad	varchar(8)	NO	PRI	NULL	
nameCiudad	varchar(40)	YES		NULL	

- **Estado:**

```
describe estado;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
idEstado	bit(1)	NO	PRI	NULL	
nameEstado	varchar(20)	YES		NULL	

- **Factura:**

```
describe factura;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
idFactura	varchar(8)	NO	PRI	NULL	
idUsuario	varchar(10)	YES	MUL	NULL	
fecha	date	YES		NULL	
codigo	varchar(15)	YES	MUL	NULL	
cantidad	int(10)	YES		NULL	
total	double	YES		NULL	
obser	varchar(250)	YES		NULL	

- **Producto:**

```
describe producto;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
codigo	varchar(15)	NO	PRI	NULL	
nameProducto	varchar(30)	YES		NULL	
precio	double	YES		NULL	
stock	int(11)	YES		NULL	
descripcion	varchar(250)	YES		NULL	
talla	varchar(3)	YES		NULL	
imagen	longblob	YES		NULL	
idEstado	bit(1)	YES	MUL	NULL	

- **Usuario:**

```
describe usuario;
```

[ [Editar en línea](#) ] [ [Editar](#) ] [ [Crear código PHP](#) ]

Opciones extra

Field	Type	Null	Key	Default	Extra
idUsuario	varchar(10)	NO	PRI	NULL	
docType	varchar(30)	NO	PRI	NULL	
firstName	varchar(30)	YES		NULL	
secondName	varchar(30)	YES		NULL	
surname	varchar(30)	YES		NULL	
secondSurname	varchar(30)	YES		NULL	
indicativo	varchar(4)	YES		NULL	
phone	varchar(15)	YES		NULL	
correo	varchar(30)	YES		NULL	
direccion	varchar(60)	YES		NULL	
pass	varchar(60)	YES		NULL	
idCiudad	varchar(8)	YES	MUL	NULL	
idEstado	bit(1)	YES	MUL	NULL	

## **ANÁLISIS DE MIGRACIÓN**

En este punto deben identificarse los cambios específicos que hay que realizar para transformar cada elemento de la base de datos MySQL en un elemento de PostgreSQL que funcione de la misma manera. También hay que identificar los cambios a realizar en el código de la propia aplicación para que funcione sobre una base de datos PostgreSQL.

El gestor de base de datos de PostgreSQL es bastante parecido en cuanto a consultas y operaciones sencillas, así que no se verá cambio en el caso de la base de datos.

## **ESTRATEGIA DE MIGRACIÓN (PLAN DE MIGRACIÓN).**

### **ELABORAR PLAN DE EJECUCIÓN**

El plan de ejecución contiene todas las tareas necesarias para pasar del sistema origen al sistema destino. Entre esas tareas está la implementación de todas las transformaciones que se hayan identificado, tanto de base de datos como de código, pero también la elaboración de todos los planes asociados a la ejecución de la migración, como pueden ser:

#### **Plan de Marcha atrás.**

Si en el proceso de migración llega a suceder algo imprevisto o por si el cliente decide no realizar la migración, es importante realizar un backup antes de empezar a desarrollar el plan de tramitación, esto es por si pasa cualquier evento que no esté contemplado, y así dar una solución rápida y óptima. Es importante tratar de no modificar el ambiente en el que está instalada la base de datos MySql, para en caso de que se diga que no se realice la migración solo sea iniciar el servicio de base de datos MySql.

Tener en cuenta que la documentación es un tema fundamental en este punto tanto de lo que había como de lo que existe actualmente. Es decir, configuraciones, usuarios, etc.

### [Plan de backup](#)

#### **Mecanismo de escalado y manejo de problemas.**

Las posibilidades de qué ocurran problemas externos como corte de luz y problemas de hardware, se deben comunicar con la persona encargada por medio del correo electrónico y número telefónico los cuales son:

Son las posibles contingencias ante problemas externos a la migración, como podrían ser cortes de luz, etc. Estas contingencias incluyen personal de contacto, listas de correo, posibilidad de reuniones de urgencia, etc.

#### **Plan de Implantación.**

Contiene toda la información de los pasos a seguir para la realización del cambio de un SGBD a otro.

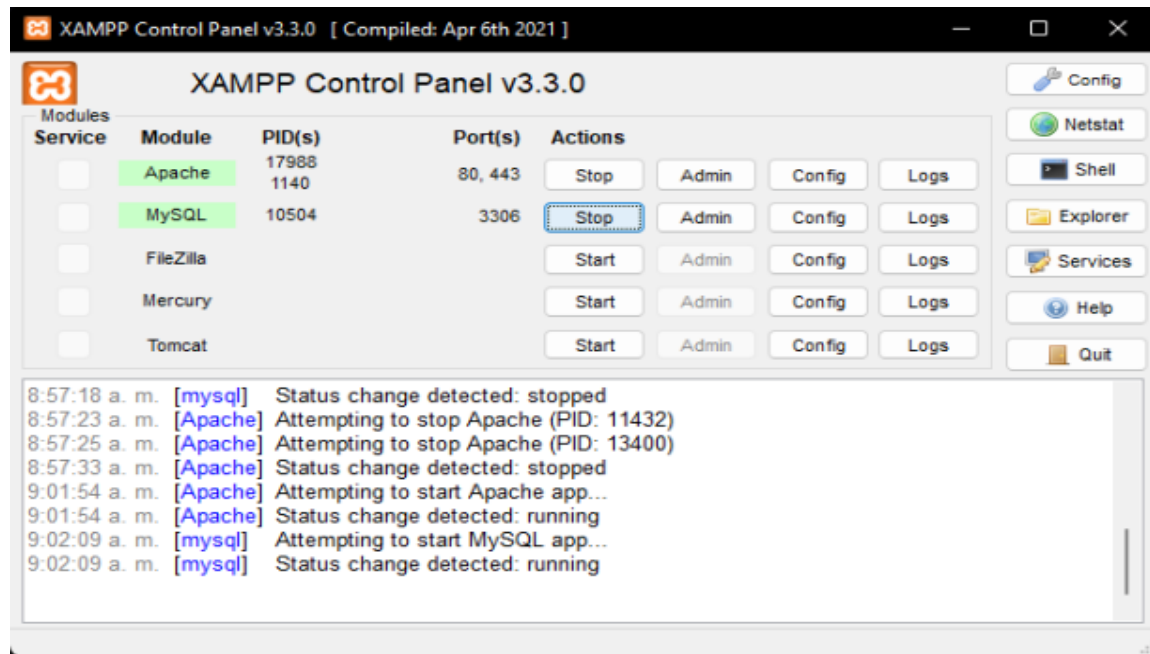
- ✓ Desglose de Tareas de cada paso (implementadores, probadores,...)
- ✓ Duración prevista de cada paso (Margen máximo de desviación)
- ✓ Flujo de los pasos (dependencias entre tareas, posibilidad de paralización de las mismas)
- ✓ Eventos o responsable que toma la decisión de éxito de la migración o marcha atrás de la misma.

En este espacio se debe anexar el informe de migración de sus proyectos grupales. En cuanto los tres tipos:

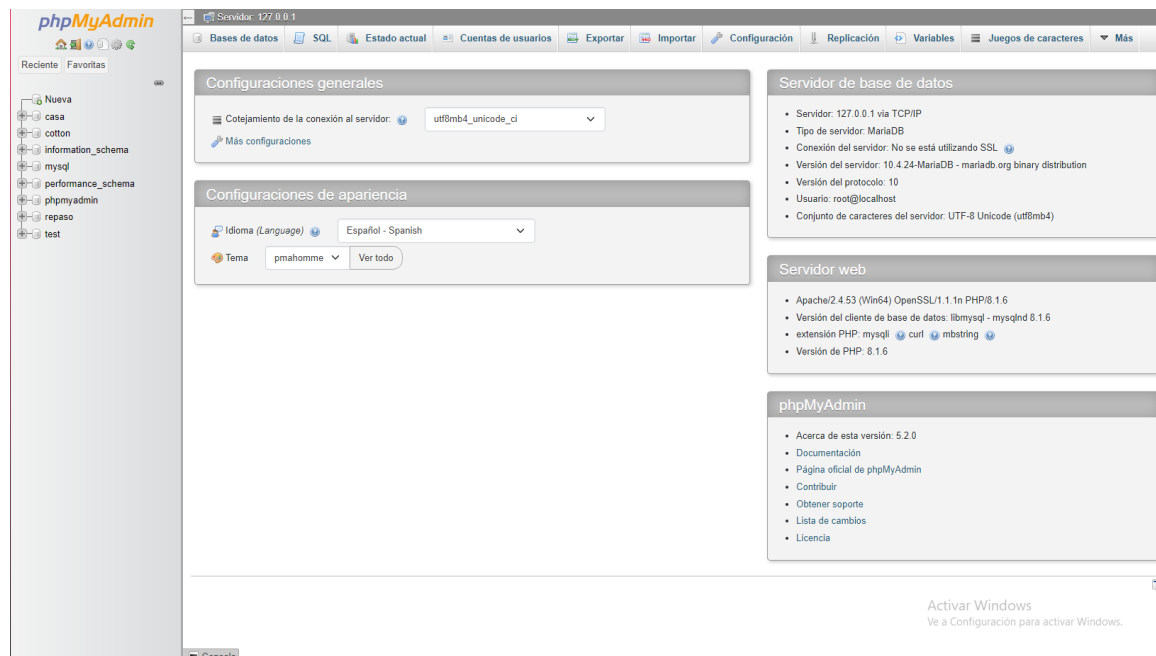
- Mismo motor – diferente ubicación local
- Mismo motor – Servidor en la Nube \* Opcional
- Diferentes motores - FullConvert (Migración DB).

#### **Mismo motor a diferente ubicación.**

Vamos a comenzar iniciando Apache y Mysql desde la consola de control de xampp



Luego nos vamos a dirigir a la página de phpMyAdmin.



Primero tenemos que tener el archivo de sql de nuestro proyecto que previamente hemos generado con lo visto en el plan de backup. Este mismo archivo igualmente lo podemos generar a través de phpMyAdmin en el apartado de exportar (tener en cuenta todo lo que se va a exportar).

## Plan de Migración de Bases de Datos

## Exportar

Exportar bases de datos del servidor actual

Exportar plantillas:

Nueva plantilla:

Plantillas existentes:

Nombre de plantilla

Nombre de plantilla

Crear

Plantilla:

-- Seleccionar plantilla --

Actualizar

Borrar

Método de exportación:

☒ Rápido - mostrar sólo el mínimo de opciones de configuración

☐ Personalizado - mostrar todas las opciones de configuración posibles

Formato:

SQL

Exportar

Ya teniendo el archivo nos vamos a dirigir a la otra ubicación e importamos el archivo, esto puede ser realizado a través de la consola con MariaDB o phpMyAdmin en el apartado de importar.

## Importar

Importando al servidor actual

Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.  
Un archivo comprimido tiene que terminar en `[formato].[compresión]`. Por ejemplo: `.sql.zip`

Buscar en su ordenador: (Máximo: 40MB)

Seleccionar archivo

cotton.sql

También puede arrastrar un archivo en cualquier página.

Conjunto de caracteres del archivo:

utf-8

Importación parcial:

☒ Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP.  
Esta puede ser una buena forma para importar archivos grandes, sin embargo, puede romper las transacciones.

Omitir esta cantidad de consultas (en SQL) desde la primera:

0

Otras opciones

☒ Habilite la revisión de las claves foráneas

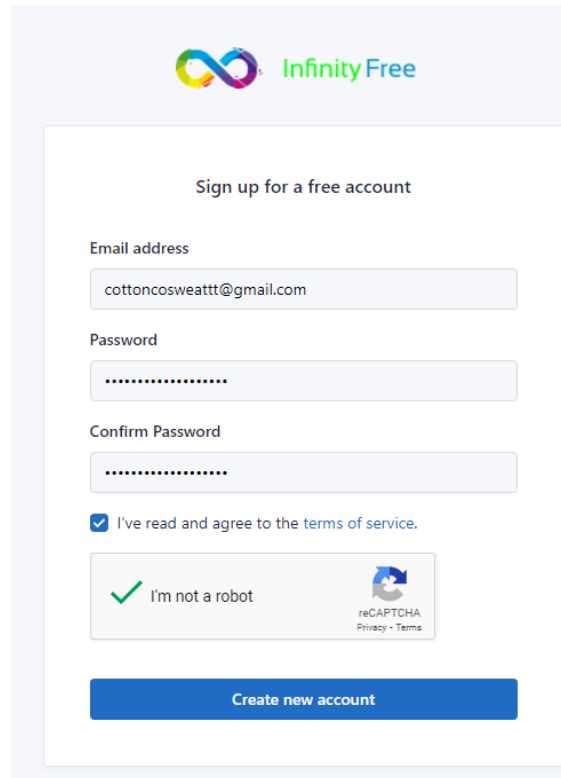
Formato

SQL

Activar Windows  
Ve a Configuración para activar Windows.

## Mismo motor a Servidor en la nube.

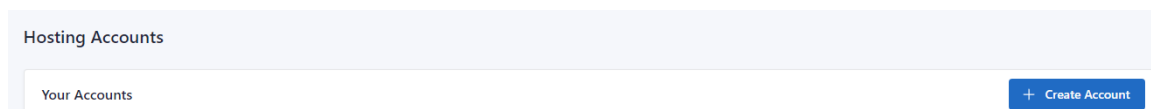
Teniendo ya el paso de exportar la base de datos o realizar el backup, vamos a dirigirnos a infinity free y vamos a crear una cuenta en Infinity Free.



The image shows the Infinity Free sign-up page. At the top is the Infinity Free logo. Below it is a heading "Sign up for a free account". The form contains the following fields and elements:

- Email address:** A text input field containing "cottoncosweatt@gmail.com".
- Password:** A password input field with masked characters ".....".
- Confirm Password:** A password input field with masked characters ".....".
- Agreement:** A checked checkbox followed by the text "I've read and agree to the [terms of service](#)".
- reCAPTCHA:** A box containing a green checkmark, the text "I'm not a robot", and the reCAPTCHA logo with links for "Privacy" and "Terms".
- Create new account:** A blue button at the bottom of the form.

Crearemos una cuenta con el botón “+Create Account”.



The image shows a section titled "Hosting Accounts". Below the title is a box labeled "Your Accounts". To the right of this box is a blue button with a plus sign and the text "+ Create Account".



Vamos a escoger un nombre de dominio y en la extensión de dominio va a quedar .epizy.com

### Step 1: Choose a Domain Name

Domain Type

Subdomain

Custom Domain

Subdomain

cottonn

Domain Extension

.epizy.com


You can add more domains after your account has been created.

Back

Check Availability

Luego vamos a ingresar a la cuenta que acabamos de crear.

### Your Accounts



[epiz 32395329](#)  
Website for cottoncosweaters.epizy.com

Active Accounts: 1 / 3

Vamos al apartado de “Control Panel”

epiz\_32395329 (Website for cottoncosweaters.epizy.com)

[Control Panel](#) [File Manager](#)

Account Details		FTP Details	
Username	epiz_32395329	FTP Username	epiz_32395329
Password	***** <a href="#">Show/Hide</a>	FTP Password	***** <a href="#">Show/Hide</a>
Status	<span>Active</span>	FTP Hostname	ftpload.net
Label	Website for cottoncosweaters.epizy.com	FTP Port (optional)	21
Main Domain	nrodbzzc.epizy.com		
Website IP	185.27.134.98		
Hosting Volume	vol16_1		
Created on	2022-08-16		

En el apartado de “Databases” nos dirigimos a la parte en donde dice “phpMyAdmin”

DATABASES

[phpMyAdmin](#) [MySQL Databases](#) [Remote MySQL](#) [PostgreSQL Databases](#)

En esta parte como podemos ver es exactamente el mismo al phpMyAdmin local a través de localhost, así que simplemente vamos a seguir el último paso del anterior apartado, importar la base en la parte de “Importar”

Servidor: sql108.epizy.com » Base de datos: epiz\_32395329\_cotton

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones Rutinas Diseñador

### Importando en la base de datos "epiz\_32395329\_cotton"

**Archivo a importar:**

El archivo puede ser comprimido (gzip, bzip2) o descomprimido.  
Un archivo comprimido tiene que terminar en **[formato].[compresión]**. Por ejemplo: **.sql.zip**

Buscar en su ordenador:  cotton.sql (Máximo: 300MB)

También puede arrastrar un archivo en cualquier página.

Conjunto de caracteres del archivo:

**Importación parcial:**

☒ Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP. *(Esto podría ser un buen método para importar archi*

Omitir esta cantidad de consultas (en SQL) desde la primera:

**Otras opciones:**

☒ Habilite la revisión de las claves foráneas

**Formato:**

Algo para tener en cuenta, como se utiliza un servidor diferente al que teníamos localmente, el archivo que se conecta a la base de datos también debe cambiarse acordemente.

```
$conexion = new mysqli("sql108.epizy.com", "epiz_32395329", "3kkAOWTENQmJR58", "epiz_32395329_cotton");  
return $conexion;
```

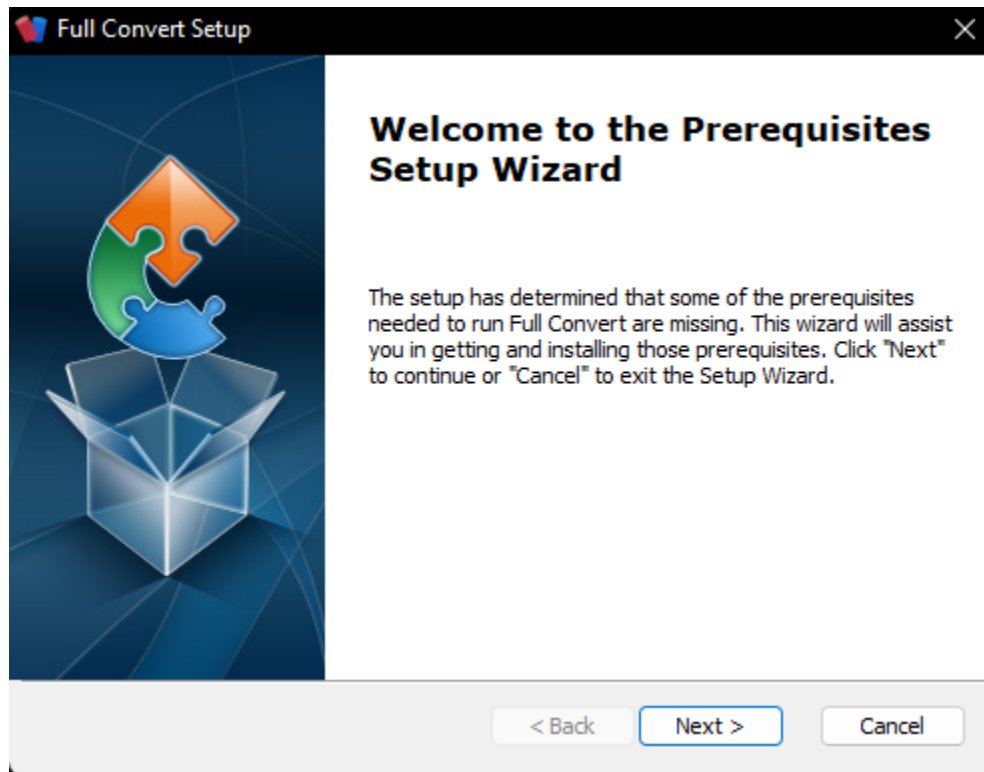
MySQL Details	
MySQL Username	epiz_32395329
MySQL Password	<div>3kkAOWTENQmJR58</div> <div>Show/Hide</div>
MySQL Hostname	sql108.epizy.com
MySQL Port (optional)	3306
Database Name	epiz_32395329_XXX (create this in the control panel)

### Diferente motor (Full Convert).

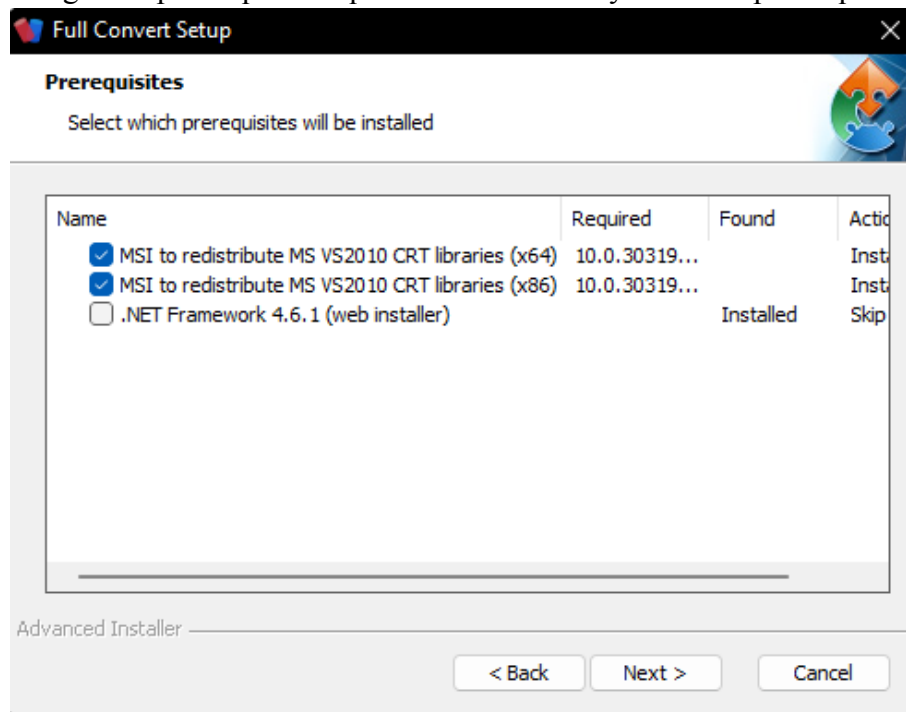
Vamos a realizar la migración a PostgreSQL con ayuda de un programa llamado FullConvert, el cual vamos a descargar con el siguiente link

[FullConvert](#)

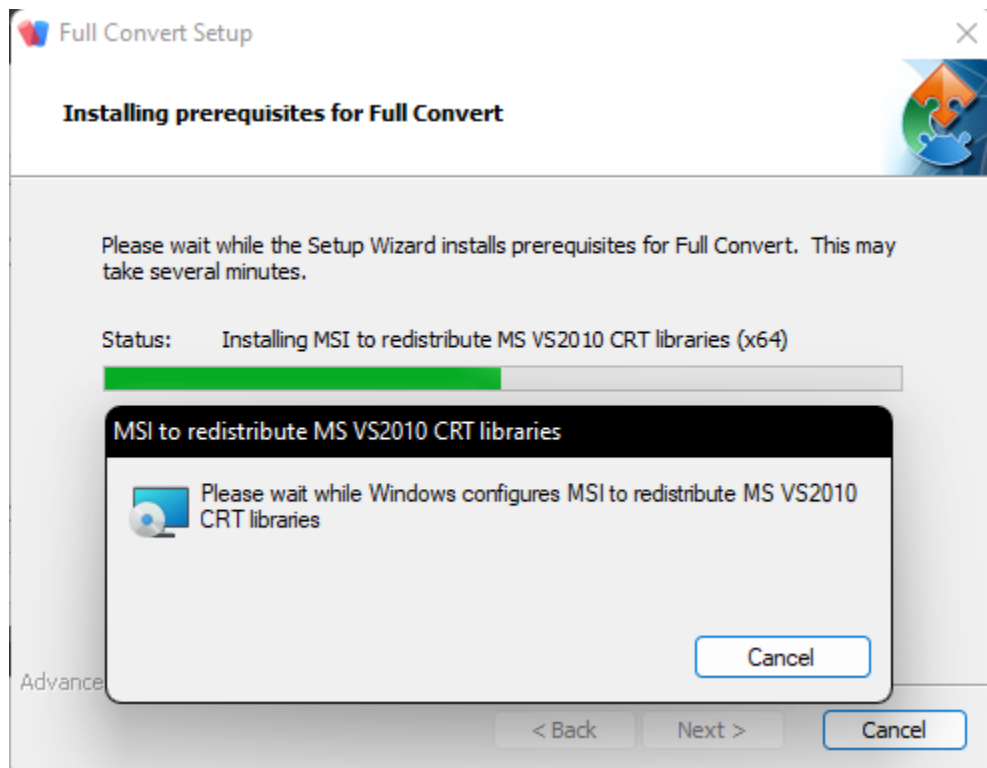
Cuando iniciemos a descargarlo nos saldrá la pantalla que nos avisa de los requisitos de instalación.



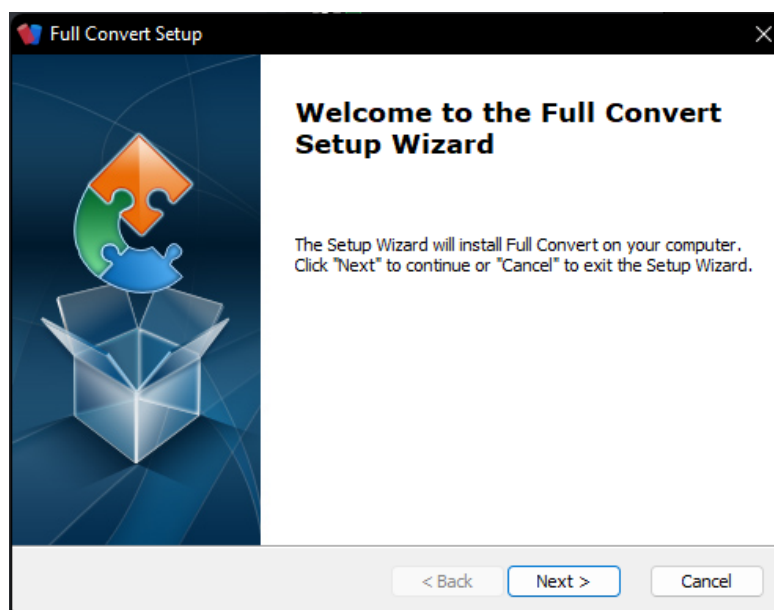
Vamos a escoger los prerequisites que inicien con MSI y obviar el prerequisite de .net.



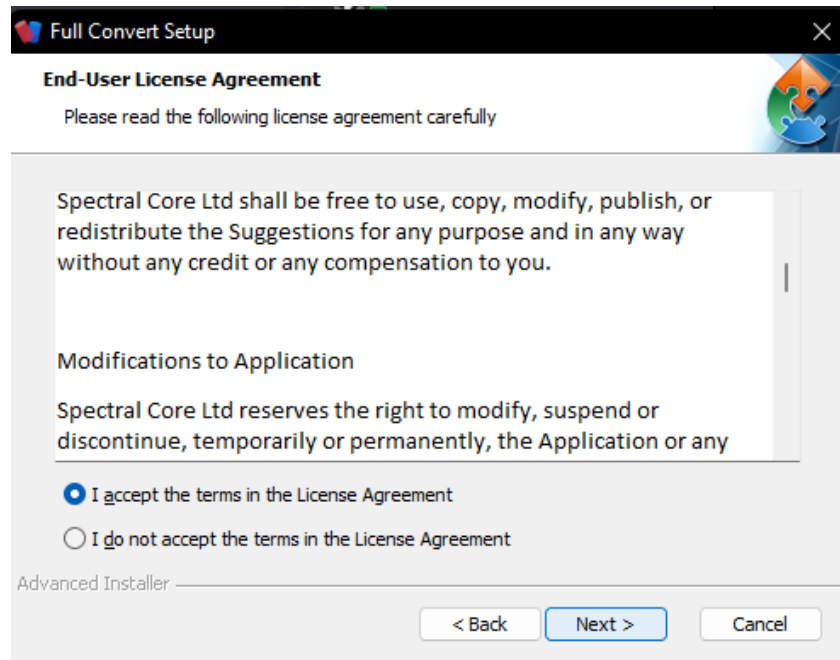
Después de seleccionarlos y darle a continuar nos saldrá la pantalla de descarga.



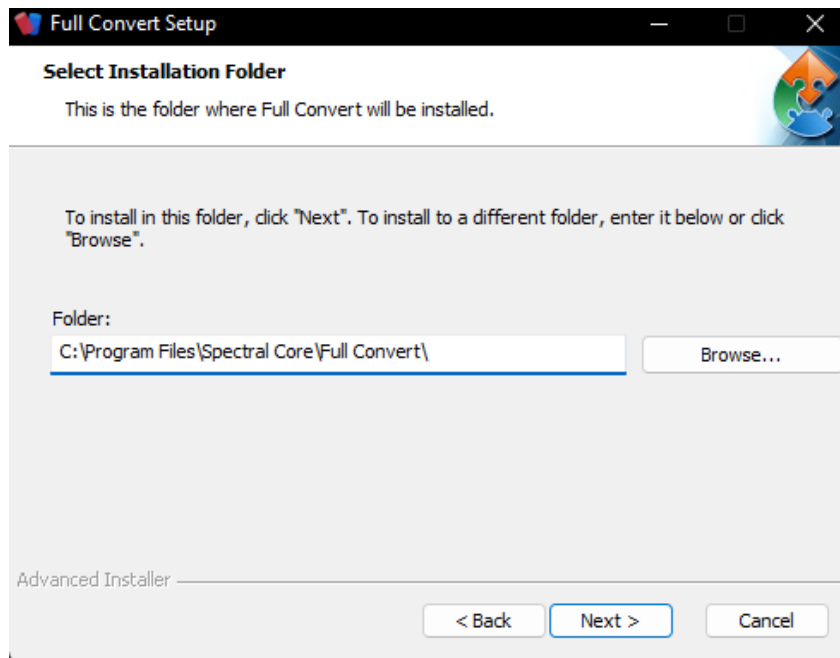
Luego de esto nos saldrá la página de inicio, le vamos a dar al botón de “next”.



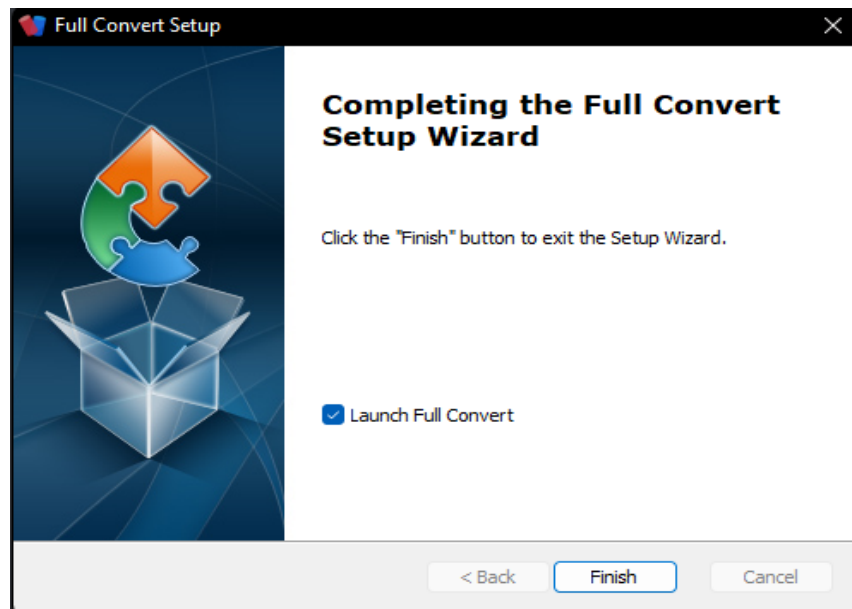
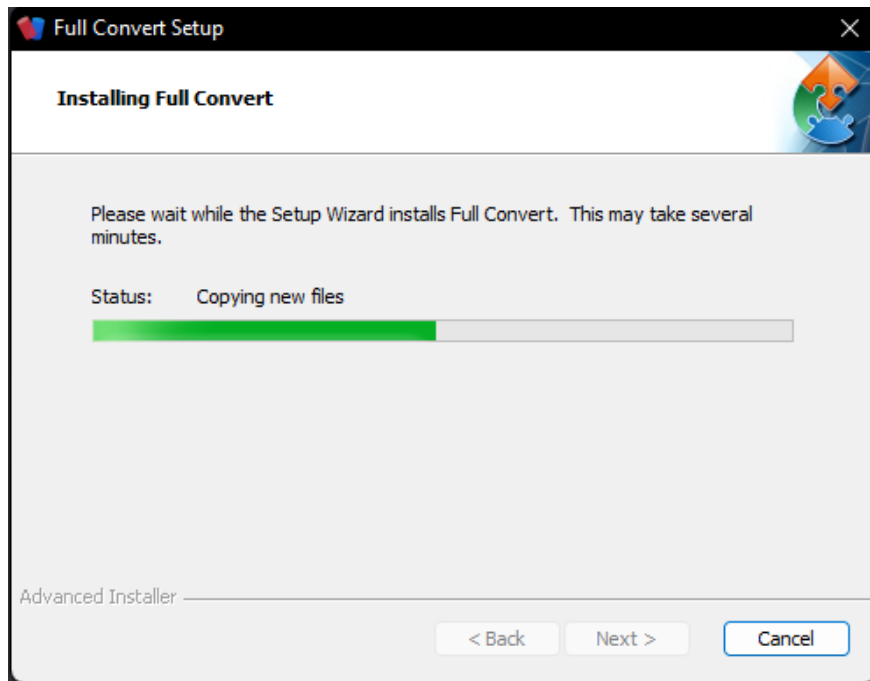
Tendremos que aceptar los términos y condiciones de la aplicación para poder descargarla.



Después vamos a seleccionar la carpeta en donde vamos a guardar la aplicación.

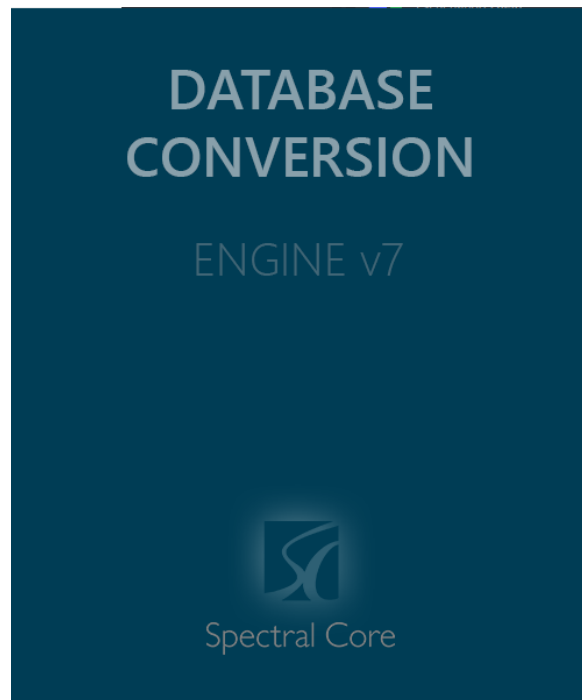


Vamos a esperar a que se instale, ya tenemos el programa FullConvert.

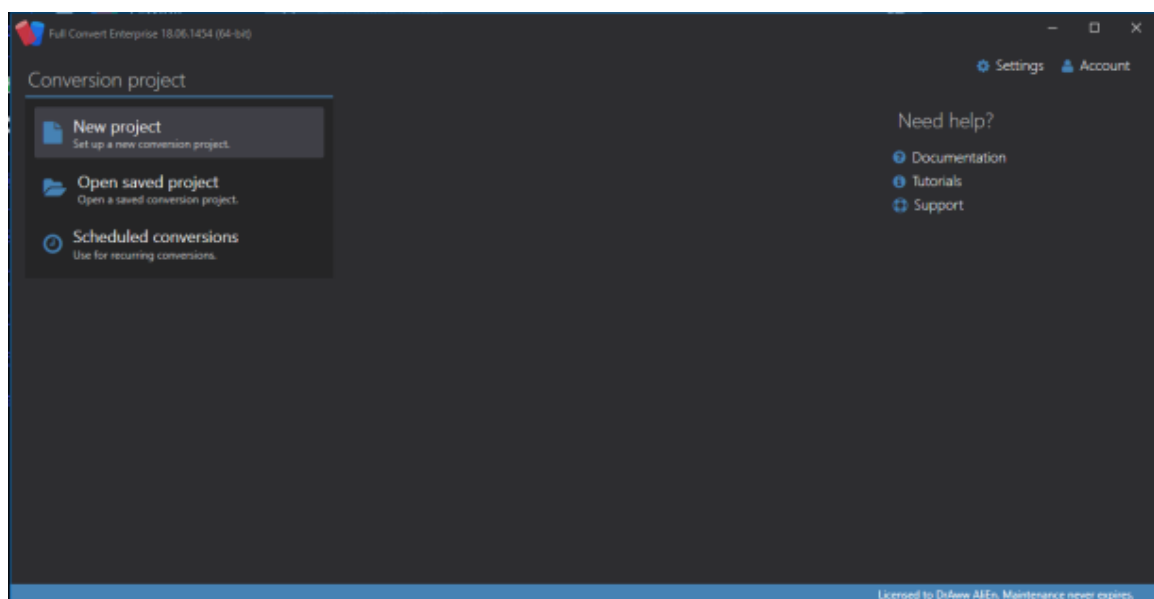




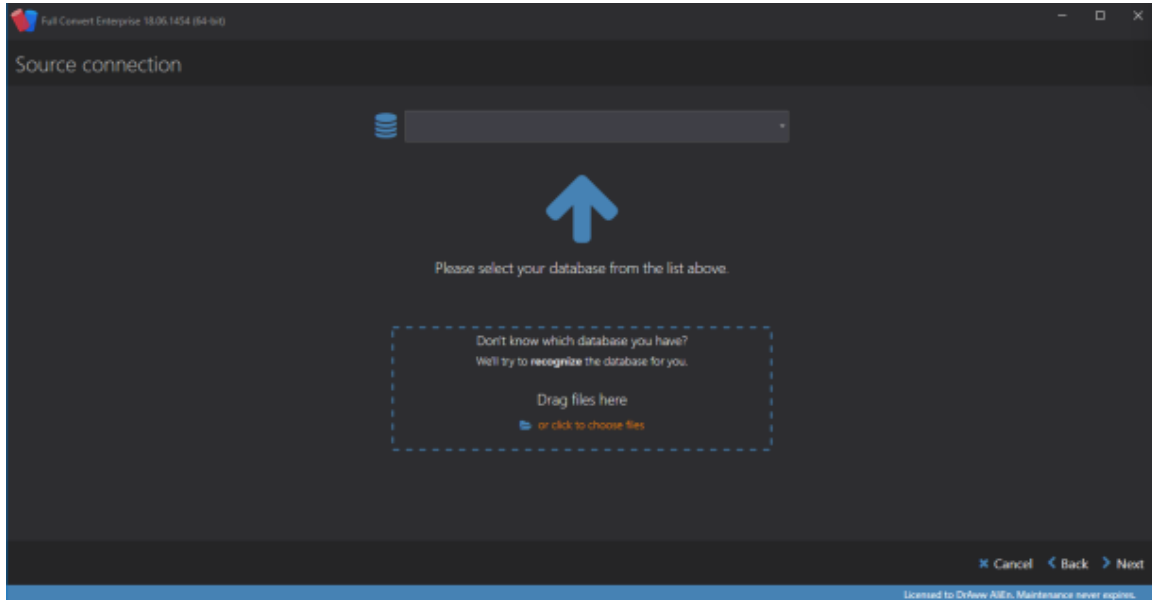
Ya con la instalación podemos abrir el programa.



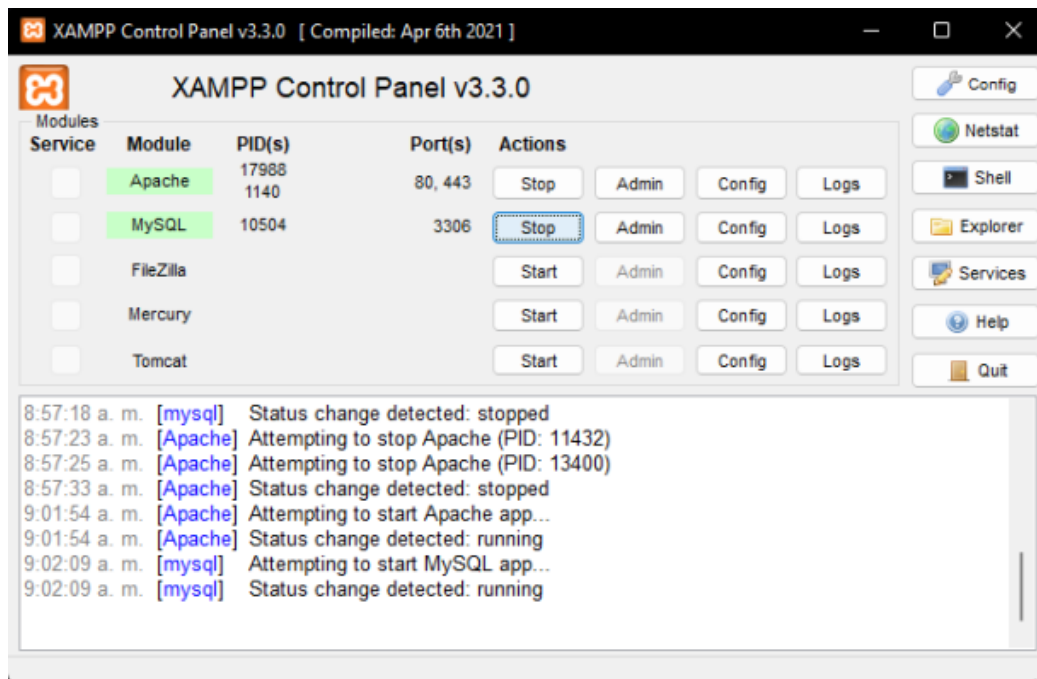
Nos saldrá una interfaz en la cual vamos a poder seleccionar los archivos que necesitamos migrar.



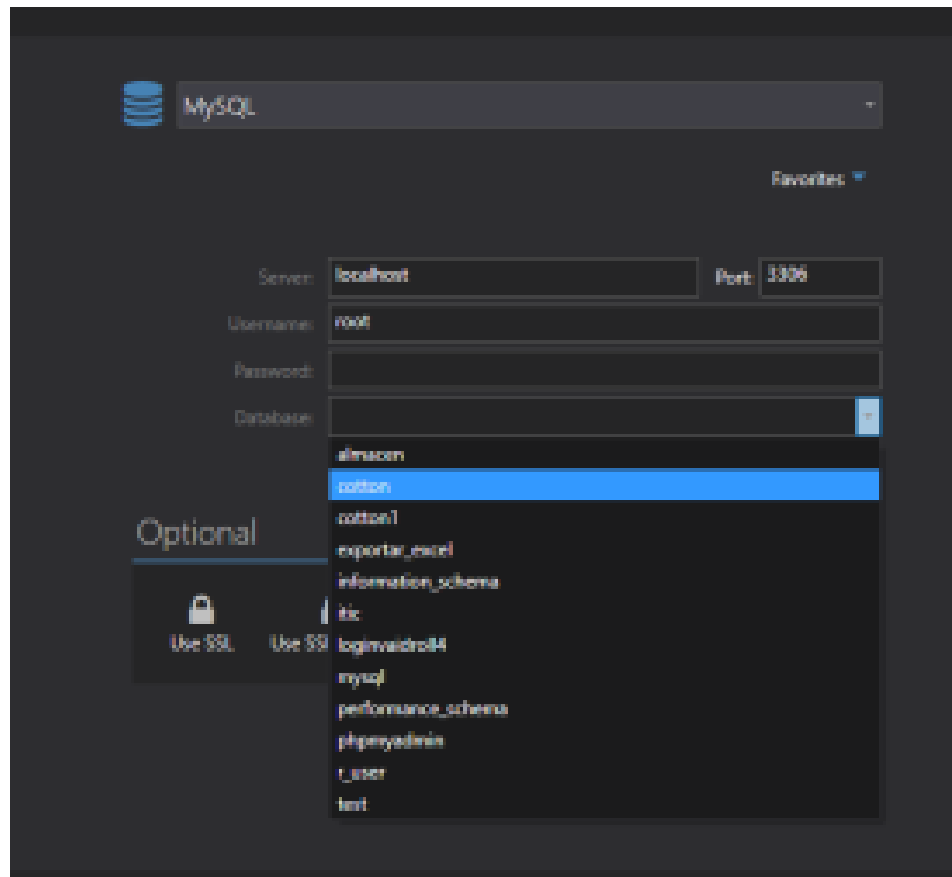
Al darle click a “New project” nos saldrá un campo en donde podemos agregar el archivo que necesitamos.



Vamos a encender Apache y MySQL, necesitamos tener el servidor del que necesitamos realizar la migración.



Después de encenderlo vamos a seleccionar el gestor de base de datos de MySQL, el servidor que en este caso es localhost, el usuario y la contraseña (que generalmente el usuario es root y no hay contraseña). Por último seleccionamos la base de datos que vamos a migrar.



Para continuar necesitamos del gestor de base de datos PostgreSQL, en el siguiente link se encuentra el paso a paso de la instalación.

[Instalación de PostgreSQL.](#)

## **BIBLIOGRAFÍA**