

**PLAN DE MIGRACIÓN: PHPHONE**

Versión: 01

Dilan Felipe García Gómez

Johan Steven Benavides Sánchez

Servicio Nacional de Aprendizaje (SENA)

Fecha: 21/05/2025

CONTENIDOS

[1. HOJA DE CONTROL 2](#_Toc1988864863)

[2. REGISTRO DE CAMBIOS 3](#_Toc2138895103)

[3. CONTROL DE DISTRIBUCIÓN 3](#_Toc115997998)

[4. INTRODUCCIÓN 3](#_Toc1814348958)

[5. DESCRIPCIÓN FUNCIONAL 4](#_Toc1525075834)

[6. REQUISITOS DE MIGRACIÓN 4](#_Toc1935113481)

[7. MODELO DE DATOS 4](#_Toc1242768650)

[8. DESCRIPCIÓN COMPLETA DE TODAS LAS TABLAS 6](#_Toc511394893)

[9. ANÁLISIS Y DISEÑO DE PROCESOS 16](#_Toc297404965)

[10. FORMATEO DE DATOS 17](#_Toc80945171)

[11. CARGA FINAL 17](#_Toc773919798)

[12. NECESIDADES DEL ENTORNO 17](#_Toc1462080123)

[13. ANEXOS 19](#_Toc21843389)

[14. GLOSARIO 19](#_Toc761511287)

[15. BIBLIOGRAFÍA 19](#_Toc1672223240)

## HOJA DE CONTROL

| **Campo** | **Valor** |
| --- | --- |
| Organismo | SecurityDatai |
| Proyecto | PHPHONE |
| Entregable | Plan de Migración y Carga Inicial |
| Autor | Dilan Felipe García Gómez |
| Versión/Edición | 01 |
| Fecha Versión | [22/05/2025] |
| Aprobado por | Dilan García |
| Fecha Aprobación | [22/05/2025] |
| No. Total de Páginas | 14 |

## REGISTRO DE CAMBIOS

| **Versión** | **Causa del Cambio** | **Responsable del Cambio** | **Fecha del Cambio** |
| --- | --- | --- | --- |
| 01 | Versión inicial | Dilan García | [22/05/2025] |

## CONTROL DE DISTRIBUCIÓN

**Nombre y Apellidos:**  
Dilan Felipe Garcia Gomez

Johan Steven Benavides Sanchez

## INTRODUCCIÓN

* **Objetivo**

Documentar el proceso de migración de la base de datos del sistema PHPHONE utilizando Flask-Migrate, este para migrar nuestra base de datos de ORM a MySQL. Posteriormente, de MySQL a Postgres

* **Equipo de Desarrollo**: Implementación de migraciones
* **DBAs**: Ejecución en producción
* **QA**: Validación post-migración

## DESCRIPCIÓN FUNCIONAL

Migración desde el mismo sistema (conversión) utilizando:

* **Flask-Migrate**: Para gestión de cambios incrementales.
* **Seed roles**: Carga de datos iniciales (roles, datos del administrador).

**Tipo de migración**:

* Estructura: Automática
* Datos: Manual scripts Python

## REQUISITOS DE MIGRACIÓN

| **Código** | **Requisito** | **Tipo** | **Entornos** |
| --- | --- | --- | --- |
| MIG-001 | Migración esquema con Flask-Migrate | MI | Todos |
| MIG-002 | Carga inicial de roles | CO | Todos |
| MIG-003 | Preservar datos existentes | CO | Todos |
| MIG-004 | Migracion base de datos de MySQL a Postgres | MI | Todos |

## MODELO DE DATOS

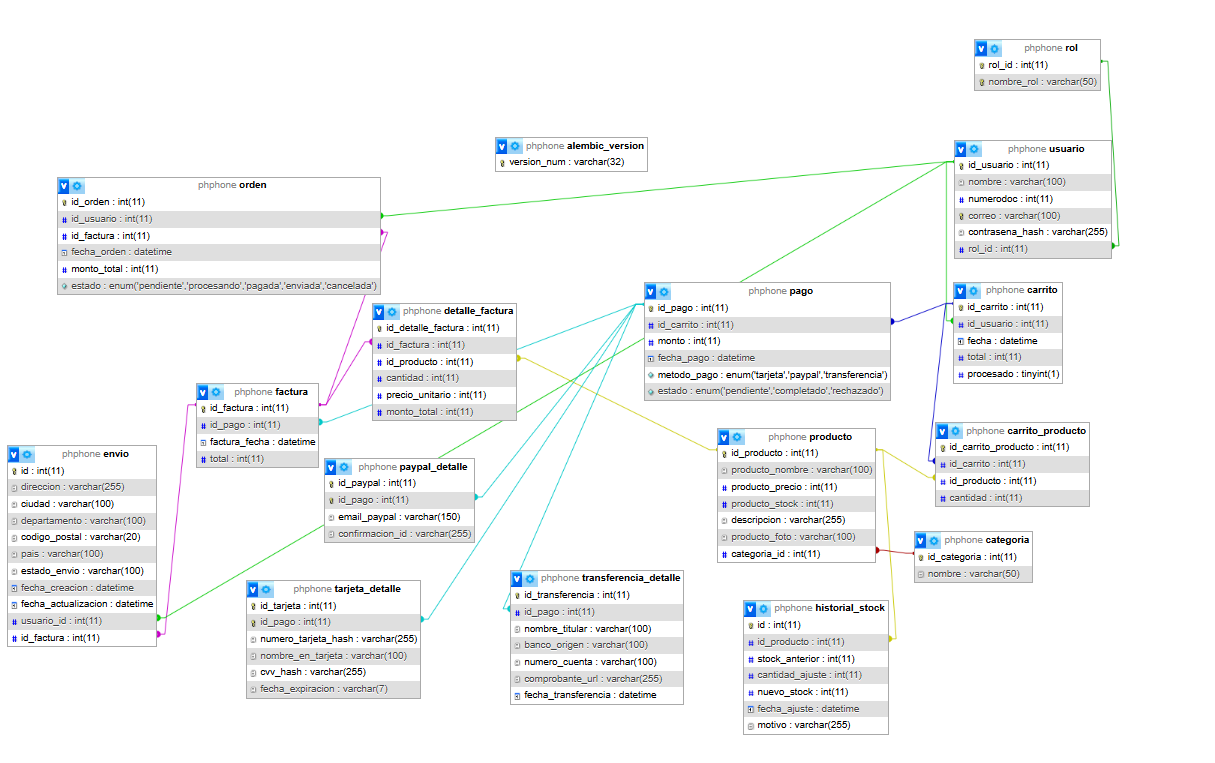
**7.1 Tablas de origen**

| **Nombre Tabla** | **Descripción** |
| --- | --- |
| **usuario** | **Almacena información de usuarios del sistema incluyendo credenciales y roles** |
| **rol** | **Define los diferentes roles de usuarios (admin, cliente, etc.)** |
| **categoria** | **Categorías para clasificar productos** |
| **producto** | **Catálogo completo de productos con información de inventario** |
| **historial\_stock** | **Registra todos los movimientos de stock de productos** |
| **carrito** | **Carritos de compra asociados a usuarios** |
| **carrito\_producto** | **Relación entre carritos y productos con cantidades** |
| **pago** | **Transacciones de pago con diferentes métodos** |
| **paypal\_detalle** | **Detalles específicos para pagos con PayPal** |
| **transferencia\_detalle** | **Información para pagos por transferencia bancaria** |
| **tarjeta\_detalle** | **Datos de tarjetas de crédito (almacenados de forma segura)** |
| **factura** | **Facturas generadas por compras** |
| **detalle\_factura** | **Items individuales que componen cada factura** |
| **orden** | **Órdenes de compra con su estado actual** |
| **envio** | **Información de envíos para las órdenes** |

**7.2 Modelo de datos final**

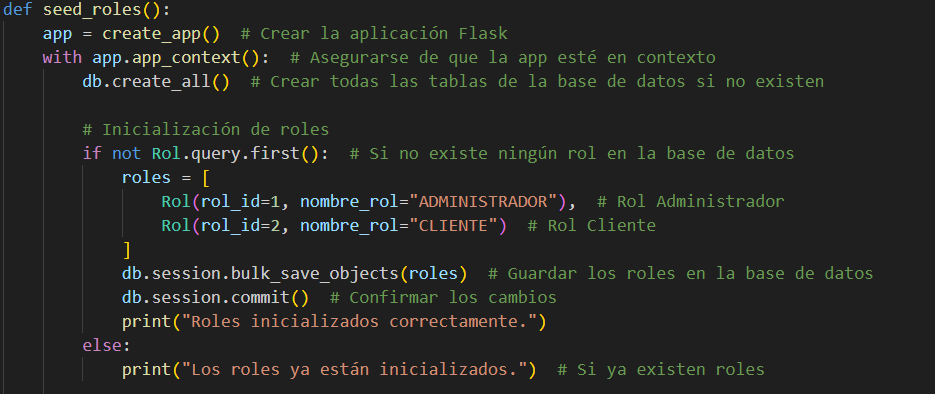
**7.2.1 Representación gráfica**

***(Diagrama ER generado con MySQL)***



## ANÁLISIS Y DISEÑO DE PROCESOS

**8.1 Carga inicial de datos**

La carga inicial de los datos se hace mediante los modelos, teniendo eso en cuenta, solo se han de ejecutar los comandos necesarios para migrar los datos.

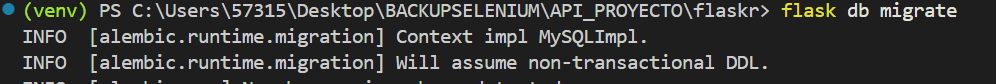
## CARGA FINAL

Comandos necesarios para ejecutar la migración:

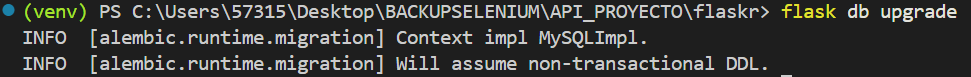
* Flask db Init: Comando para inicializar la base de datos en mysql.



* Flaks db migrate: Comando para que los modelos se migren a la base de datos:



* Flask db upgrade: Comando para actualizar la base de datos con la migracion correspondiente:

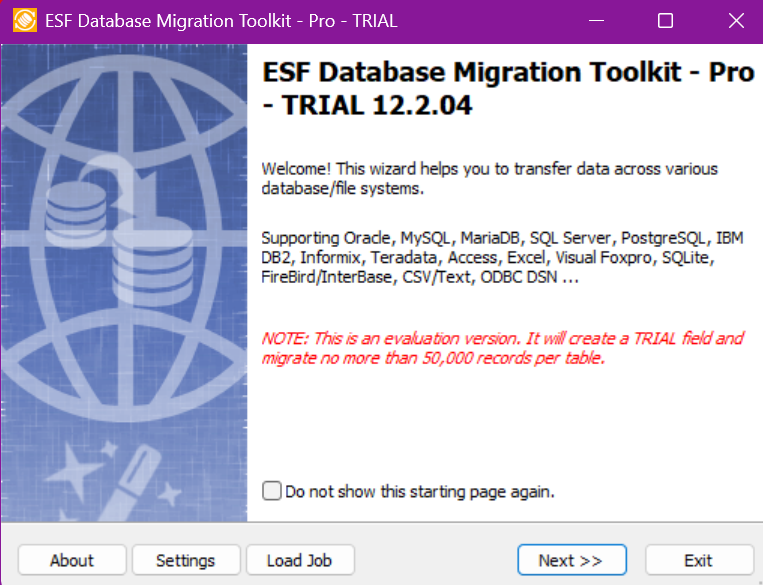


## **Migración de MySQL a Postgres**

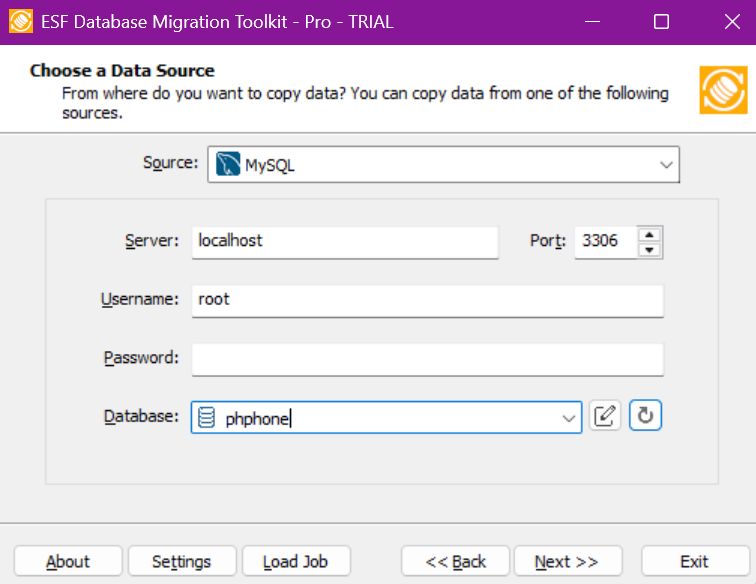
10.1 Motivo: El motivo de la migración de la base de datos se basa en el punto de que el Sistema luego de su etapa de producción necesitará un mejor gestor de su base de datos, con ello, se justifica el motivo del cambio del gestor de la base de datos, sabiendo que Postgres es un sistema algo más robusto y avanzado.

10.2 Ejecución de la Migración: Para realizar una correcta migración de nuestra base de datos:

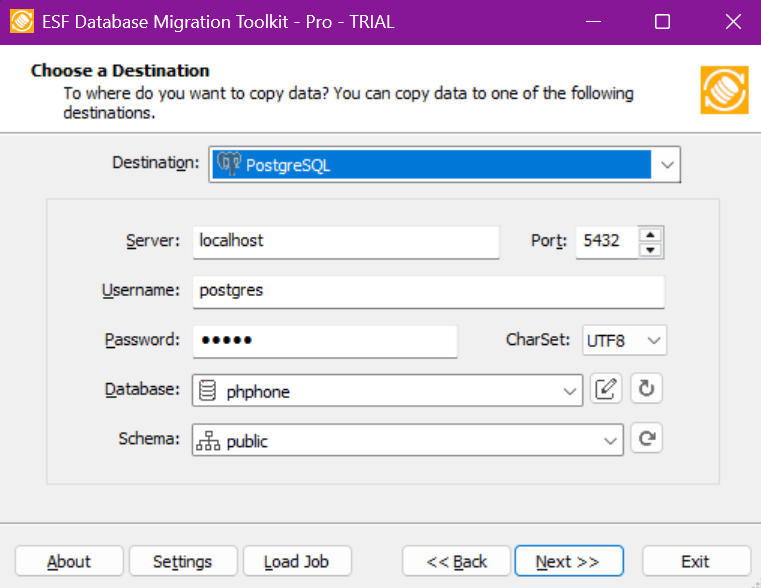
* Abrimos ESF Database Migration Doolkit:



* Clic en Next
* Configuramos de la siguiente manera:



* Clic en Next
* Configuramos de la siguiente manera:



* Nota: En password debe de ir la contraseña que establecimos al instalar ESF Database Doolkit, para ver la instalación de este, puede hacerlo en el siguiente link: [Instalacion](https://soysena-my.sharepoint.com/:w:/g/personal/jsbenavides1403_soy_sena_edu_co/EUgm-FQnrmJFu3Jdcn59IMEBr23632yaG2QDdJZPledSFw?e=uvSBYE)

10.3 Luego de realizar la migración, verificaremos la integridad de nuestras tablas y datos haciendo una comparación entre MySQL y Postgres.

## NECESIDADES DEL ENTORNO

**11.1 Hardware**

| Recurso | Especificación | Entorno |
| --- | --- | --- |
| Servidor BD | 4vCPU, 8GB RAM, 100GB SSD | Producción |

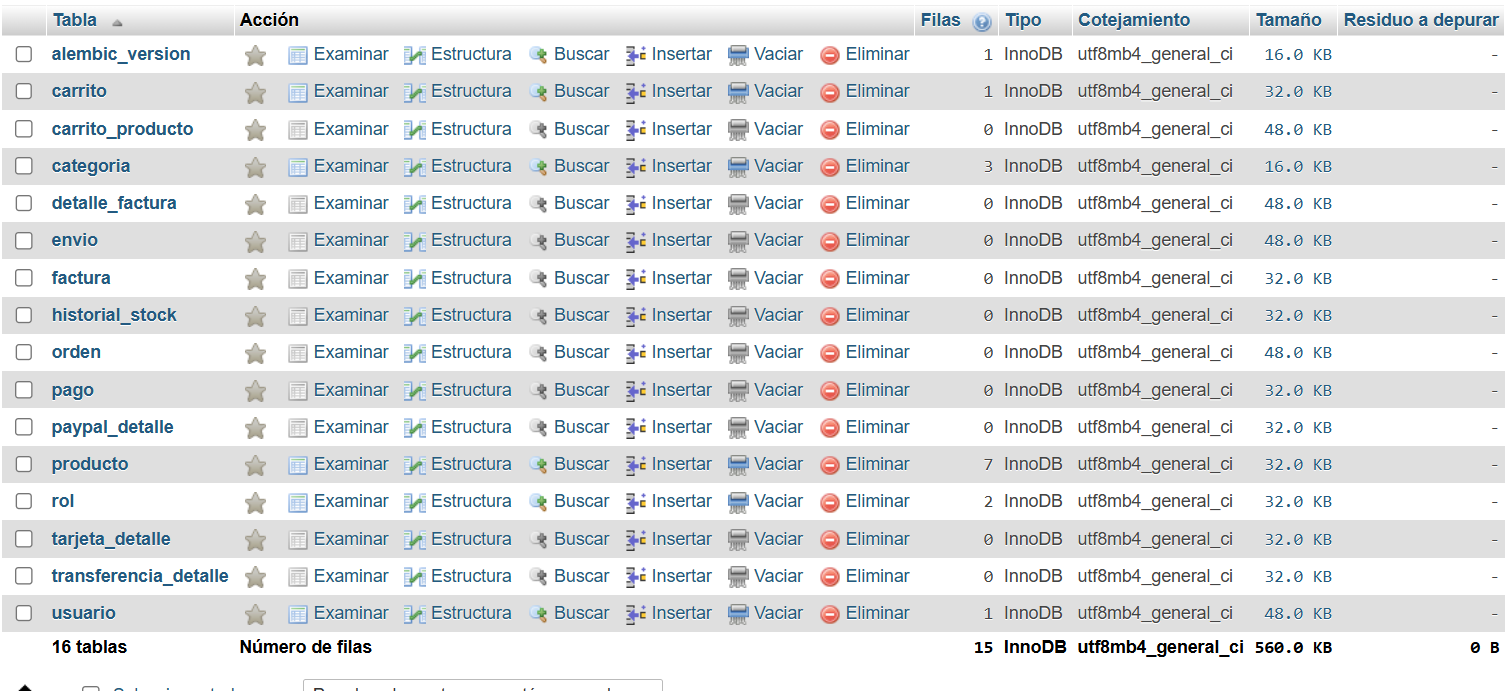
**11.2 Software**

* alembic==1.14.1
* aniso8601==10.0.0
* blinker==1.9.0
* click==7.1.2
* colorama==0.4.6
* Flask==2.0.1
* Flask-Cors==5.0.0
* Flask-JWT-Extended==4.7.1
* Flask-Migrate==3.1.0
* Flask-MySQLdb==2.0.0
* Flask-RESTful==0.3.10
* Flask-SQLAlchemy==2.5.1
* greenlet==3.1.1
* itsdangerous==2.2.0
* Jinja2==3.1.5
* Mako==1.3.9
* MarkupSafe==2.1.1
* marshmallow==3.23.0
* marshmallow-sqlalchemy==0.26.1
* mysql==0.0.3
* mysqlclient==2.2.7
* packaging==24.2
* pip==25.0.1
* PyJWT==2.10.1
* PyMySQL==1.1.1
* pytz==2025.1
* setuptools==75.8.0
* six==1.17.0
* SQLAlchemy==1.4.39
* typing\_extensions==4.12.2
* Werkzeug==2.0.3
* Flask-Mail==0.9.1

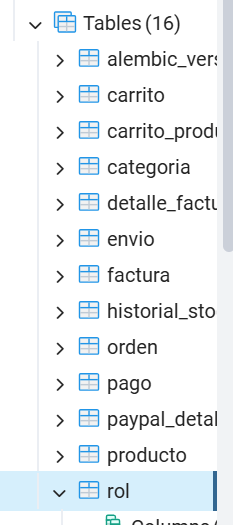
## ANEXOS

**Script de validación post-migración:**

* Tablas en MySQL:



* Tablas en Postgres:



## GLOSARIO

* **Flask Migrate**: Herramienta de migraciones para SQLAlchemy
* **Seed**: Datos iniciales requeridos por el sistema

## BIBLIOGRAFÍA

* Documentación oficial de Flask-Migrate
* SQLAlchemy 1.4 Documentation