



GERO Y NATIS

Plan de Instalación

Daniel Felipe Bermeo Torres

Deisy Catherine Gonzalez Bueno

Jennifer Vanessa Mateus Farfan

ADSO – 2900602

Versión 01

Fecha: 01/06/2025

HOJA DE CONTROL

Organismo	SENA – Centro de Electricidad, Electrónica y Telecomunicaciones		
Proyecto	Gero y Natis		
Entregable	Plan de Instalación		
Autor	Daniel Felipe Bermeo Torres Deisy Caterine González Bueno Jennifer Vanessa Mateus Farfán		
Versión/Edición	0100	Fecha Versión	01/06/2025
Aprobado por		Fecha Aprobación	10/06/2025
		Nº Total de Páginas	24

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0100	Versión inicial	Daniel Felipe Bermeo Torres	01/06/2025

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Daniel Felipe Bermeo Torres



ÍNDICE

1	INTRODUCCIÓN	5
1.1	Descripción del proyecto	5
1.2	Objetivos	5
1.3	Alcance	5
1.3.1	Áreas/departamentos involucrados:	5
1.3.2	Número de usuarios:	6
1.3.3	Sedes / Ubicaciones	6
1.3.4	Módulos / Componentes a Instalar	6
1.3.5	Exclusiones	6
2	REQUISITOS	7
2.1	Requisitos de Hardware	7
2.2	Requisitos de Software	7
2.3	Requisitos de infraestructura.....	8
2.3.1	Servidores	8
2.3.2	Base de datos:.....	8
2.3.3	Conectividad:	9
2.3.4	Almacenamiento	9
2.3.5	Servicios Cloud.....	9
2.4	Requisitos de seguridad	9
3	ENTORNO DE INSTALACIÓN.....	9
3.1	Arquitectura de la solución	9
3.2	Entornos requeridos.....	10
3.3	Integración con sistemas existentes	10
4	PROCESO DE INSTALACIÓN.....	11
4.1	Resumen de actividades principales.....	11
4.2	Procedimiento detallado de instalación	11
4.2.1	Preparación del entorno	11
4.2.2	Instalación de prerrequisitos	11
4.2.3	Instalación del software base.....	12
4.2.4	Configuración inicial.....	13



4.3	Configuraciones específicas.....	13
4.4	Personalización	14
4.5	Migración de datos	15
5	CRONOGRAMA.....	15
5.1	Calendario de actividades	15
5.2	Hitos clave.....	15
6	Pruebas.....	16
6.1	Estrategia de pruebas	16
6.2	Tipos de pruebas a realizar	17
7	EQUIPO DE RESPONSABILIDADES.....	18
7.1	Estructura del equipo	18
7.2	Matriz RACI.....	19
7.3	Escalamiento.....	20
8	GESTIÓN DE RIESGOS.....	20
8.1	Matriz de riesgos	20
9	CAPACITACIÓN Y SOPORTE	22
9.1	Plan de capacitación	22
9.2	Soporte post-implementación.....	23



1 INTRODUCCIÓN

1.1 Descripción del proyecto

El proyecto *Gero y Natis* busca instalar un sistema web de gestión para optimizar los procesos de inventario y ventas de la empresa. La iniciativa responde a la necesidad de superar las dificultades actuales en el control de existencias, registro de productos y manejo de transacciones.

Con esta instalación se espera automatizar entradas y salidas de productos, controlar el stock en tiempo real, emitir comprobantes de venta y generar reportes útiles para la toma de decisiones. El sistema está orientado a ser utilizado por el personal de ventas, almacén y administración de la empresa.

1.2 Objetivos

- Implementar un sistema web para gestionar de forma centralizada los procesos de inventario y ventas de la empresa Gero y Natis.
- Automatizar tareas críticas como el registro de productos, control de existencias, generación de facturas y gestión de usuarios.
- Reducir errores manuales, mejorar la eficiencia operativa y minimizar tiempos en procesos clave.
- Proporcionar reportes confiables que faciliten la toma de decisiones estratégicas.
- Aumentar la productividad del personal con una herramienta intuitiva, accesible desde diferentes dispositivos.

1.3 Alcance

El sistema de gestión para el proyecto *Gero y Natis* cubrirá los procesos esenciales de control de inventario y ventas, así como las funcionalidades necesarias para el funcionamiento eficiente de las operaciones comerciales. A continuación, se detallan los principales elementos del alcance:

1.3.1 Áreas/departamentos involucrados:

- Área de Ventas
- Área de almacén / Administrador



1.3.2 Número de usuarios:

El sistema será utilizado inicialmente por 6 usuarios, con posibilidad de expansión. Cada usuario tendrá un rol específico:

- **Vendedores:** Con acceso exclusivo al módulo de ventas, utilizado en el punto de caja.
- **Administradores:** Con acceso total a todos los módulos del sistema, incluyendo configuración, reportes, productos, usuarios y seguridad.

1.3.3 Sedes / Ubicaciones

- **Tienda física principal:** Donde se realizan las ventas presenciales y se encuentra instalada la caja. Aquí los vendedores acceden únicamente al módulo de ventas.
- **Área administrativa:** Ubicada dentro de la misma tienda, desde donde los administradores acceden a todos los módulos del sistema a través de la web.

1.3.4 Módulos / Componentes a Instalar

- **Módulo Productos:** Registro, Edición, Inhabilitación, búsqueda
- **Módulo de Inventario:** Entradas, Salidas, Control de stock.
- **Módulo de Ventas:** Registro, Historial de clientes, Calendario.
- **Módulo de usuarios:** Gestión de cuentas, Roles y Permisos.
- **Módulo de Proveedores:** Registro, Edición.

1.3.5 Exclusiones

Este proyecto se centrará exclusivamente en el desarrollo e implementación de un sistema de gestión para inventario y ventas dentro del entorno operativo de *Gero y Natis*. Por lo tanto, se excluyen expresamente los siguientes aspectos:

- **Integración con sistemas externos o de terceros** (por ejemplo, plataformas de pago en línea, sistemas contables, pasarelas de facturación electrónica u otros ERP), salvo que se especifique lo contrario en futuras fases del proyecto.
- **Desarrollo de funcionalidades que no estén directamente relacionadas** con los procesos clave definidos, tales como módulos de recursos humanos, nómina, contabilidad financiera u otras áreas administrativas no vinculadas a inventario o ventas



2 REQUISITOS

2.1 Requisitos de Hardware

Para garantizar un funcionamiento óptimo del sistema de gestión de inventario y ventas de Gero y Natis, se establecen los siguientes requisitos de hardware tanto para estaciones de trabajo como para el servidor (en caso de instalación local)

Componente	Requisitos mínimos	Requisitos Recomendados
Procesador	Intel Core i3	Intel Core i5 o AMD Ryzen 3
Memoria RAM	4 GB	8 GB o superior
Almacenamiento	50 GB	125 GB o más, con SSD preferiblemente
Tarjeta Gráfica	Integrada	Integrada o dedicada (opcional)
Red	Ethernet 50 Mbps	Ethernet 100–200 Mbps o superior

2.2 Requisitos de Software

El sistema de gestión de inventario y ventas de Gero y Natis será desarrollado utilizando tecnologías web ampliamente adoptadas y de código abierto, como PHP, MySQL y Bootstrap. Además, se utilizará Composer para la gestión de dependencias, incluyendo librerías como PHPMailer y Cloudinary, lo que permite una estructura modular, mantenible y adaptable.

Software	Versión	Propósito
PHP	v8.2	Lenguaje de programación principal para la lógica del sistema (backend).
MySQL	v5.7 o superior	Sistema de gestión de base de datos relacional



		para almacenar la información.
Apache	v2.4 o superior	Servidor web utilizado para alojar y ejecutar la aplicación.
Composer	v2.8.8	Gestor de dependencias usado para integrar librerías como PHPMailer y Cloudinary.
Bootstrap	v5.3	Framework CSS para la construcción de una interfaz responsive y amigable.
PHPMailer	v6.x	Librería utilizada para el envío de correos desde el sistema.
Cloudinary PHP	Última versión	Servicio utilizado para la carga y gestión de imágenes en la nube.

2.3 Requisitos de infraestructura

Para garantizar la estabilidad, disponibilidad y escalabilidad del sistema de gestión de inventario y ventas de Gero y Natis, se requieren las siguientes condiciones de infraestructura:

2.3.1 Servidores

Durante el desarrollo y pruebas, el sistema se ejecuta en entornos locales mediante XAMPP (Apache + PHP + MySQL). Para la etapa de producción, se contempla el despliegue en Microsoft Azure, utilizando una instancia de máquina virtual o App Service con soporte para PHP 8.2.

2.3.2 Base de datos:

El sistema utiliza MySQL 5.7 o superior. En desarrollo, se usa una base local; en producción, se usará una instancia administrada en Azure Database for MySQL, con respaldo automático y acceso restringido por IP.



2.3.3 Conectividad:

Se requiere conexión a Internet estable para acceder a la aplicación en Azure, además de red local para el uso interno en la tienda. Se recomienda una conexión mínima de 10 Mbps simétricos para el servidor y puntos de venta.

2.3.4 Almacenamiento

En desarrollo local, se recomienda al menos 50 GB de espacio libre. En producción, se utilizará almacenamiento escalable en la nube (Azure Storage) para logs, respaldos y archivos temporales. Las imágenes de productos se almacenarán externamente mediante Cloudinary.

2.3.5 Servicios Cloud

- **Azure** (Hosting y base de datos)
- **Cloudinary** (Gestión de imágenes)
- **PHPMailer** (envío de correos)

2.4 Requisitos de seguridad

La seguridad de los datos y del acceso al sistema es un aspecto fundamental del proyecto Gero y Natis. Para ello se establecen los siguientes requisitos:

- **Permisos requeridos:**
El sistema debe permitir la gestión de usuarios mediante roles, asegurando que cada persona acceda únicamente a las funcionalidades autorizadas.
- **Autenticación:**
Implementación de un sistema de login robusto con validación de credenciales, protección contra accesos no autorizados y bloqueo de sesiones inactivas.
- **Protección de datos:**
Se implementarán copias de seguridad periódicas automáticas y cifrado de datos críticos, tanto en tránsito como en almacenamiento.

3 ENTORNO DE INSTALACIÓN

3.1 Arquitectura de la solución

La solución estará basada en una arquitectura cliente-servidor, donde:

- El servidor alojará la aplicación web desarrollada en PHP puro, utilizando Apache como servidor web y MySQL como sistema de base de datos. El



sistema se ejecutará inicialmente en entornos locales (XAMPP) y luego será desplegado en la nube (Azure).

- Los clientes accederán al sistema a través de navegadores web estándar (Google Chrome, Mozilla Firefox, etc.) desde computadoras conectadas a la red local o a Internet, dependiendo del entorno.

Esta arquitectura permite que la solución sea centralizada, escalable y fácil de mantener, facilitando el acceso multiusuario y el control centralizado de los datos.

3.2 Entornos requeridos

Entorno	Propósito	Configuración	Ubicación
Desarrollo	Programar y probar nuevas funcionalidades sin afectar usuarios reales	XAMPP, PHP 8.2, MySQL local, Bootstrap, PHPMailer, Cloudinary, Compose	Computadoras de los desarrolladores
Pruebas	Validar funcionalidades antes de subirlas al entorno real	Mismo entorno que producción, con datos de prueba	Servidor local aislado o entorno espejo en Azure
Producción	Sistema en funcionamiento real con acceso de usuarios finales	Hosting en Azure, PHP 8.2, MySQL en Azure, Apache, integración con Cloudinary y PHPMailer vía SMTP	Nube (Microsoft Azure)

3.3 Integración con sistemas existentes

Sistema	Tipo Integración	Complejidad
Cloudinary	Carga y gestión de imágenes mediante API	Baja
PHPMailer	Envío de correos usando servidor SMTP	Baja



4 PROCESO DE INSTALACIÓN

4.1 Resumen de actividades principales

- Preparación del entorno
- Instalación de prerequisites
- Instalación de software base
- Configuración inicial

4.2 Procedimiento detallado de instalación

4.2.1 Preparación del entorno

- **Verificar requisitos técnicos del sistema:**
Confirmar que el equipo o servidor donde se instalará el sistema Gero y Natis cumple con los requisitos mínimos y recomendados de hardware, software y red. Esto incluye procesador, memoria RAM, almacenamiento disponible y conectividad a Internet.
- **Adquirir o habilitar el servidor:**
En caso de que el despliegue sea en un entorno productivo, se debe contar con un servidor dedicado o VPS con acceso SSH. Si el despliegue es local o en fase de desarrollo, se puede utilizar XAMPP en un entorno de escritorio.
- **Establecer conexión a Internet y red local:**
Verificar que la máquina tenga acceso a una conexión estable a Internet y, si se requiere trabajo colaborativo o multiusuario, conexión a la red local.

4.2.2 Instalación de prerequisites

1. Instalar XAMPP:

Descargar e instalar el paquete XAMPP que incluye Apache, PHP 8.2 y MySQL desde: <https://www.apachefriends.org>.

Asegurarse de activar los servicios de Apache y MySQL desde el panel de control.

2. Instala Visual Studio Code

Descargar desde <https://code.visualstudio.com>.

Se recomienda instalar extensiones útiles como:

- PHP Extension Pack
- GitLens



- Composer Extension
- DotENV

3. Instalar Git

Descargar e instalar Git desde <https://git-scm.com>.

Verificar la instalación abriendo una terminal y ejecutando:
`git --version`

4. Instalar Composer

Descargar e instalar Composer (gestor de dependencias para PHP) desde: <https://getcomposer.org/download>.

Verificar su instalación ejecutando `composer -v` en la terminal.

5. Clonar Repositorio del sistema

Abrir una terminal y ejecutar:

```
git clone https://github.com/DanielBermeoTO/GeroyNatis2.git  
cd GeroyNatis2
```

6. Abrir proyecto en Visual Studio Code

Desde el apartado que tiene Visual Studio Code se puede abrir la carpeta clonada de github.

7. Instalar dependencias del proyecto

En una terminal con la ruta raíz del proyecto escribir:
`composer install`

Y se instalarán todas las dependencias del `composer.json`

8. Configurar archivo .env

Se va a crear un archivo con el nombre “.env” en donde deberá agregar variables como:

- Base de datos (host, usuario, contraseña, nombre)
- Cloudinary (API_KEY, API_SECRET)

4.2.3 Instalación del software base

1. Configurar la base de datos

Iniciar el módulo de MySQL desde XAMPP. Acceder a phpMyAdmin desde <http://localhost/phpmyadmin>. Crear una nueva base de datos llamada `geroynatis`.



Importar el archivo .sql ubicado en la carpeta del proyecto (si aplica).

2. Configurar archivo .env

Al poner las variables exactas de la base de datos, esta ya funcionara, ya que internamente del proyecto hay un archivo de conexión a base de datos pero sin variables, pues son las que se ponen en el archivo .env.

3. Probar el sistema en el navegador

- Iniciar Apache desde XAMPP.
- Acceder a la app desde <http://localhost/GYN/public/index.html>
- Comprobar que carga correctamente la página de inicio o login.
- Validar que se puede conectar con la base de datos.

4.2.4 Configuración inicial

- **Verificar carga automática de datos y configuración**

Al ejecutar el proyecto en XAMPP (<http://localhost/GYN/public/index.html>), el sistema ya se conecta con la base de datos usando el archivo de conexión y el .env previamente configurado.

- **Validar roles y accesos**


Acceder con un usuario de prueba y confirmar que el sistema restringe correctamente el acceso a los módulos según el rol (administrador o vendedor).

- **Realizar prueba funcional rápida**

Navegar por los módulos principales (productos, ventas, reportes) para confirmar que la estructura inicial está activa y sin errores críticos.

4.3 Configuraciones específicas

Componente	Parámetro	Valor	Justificación
Base de datos	DB_HOST	localhost	Dirección del servidor de base de datos en entorno local.
Base de datos	DB_NAME	geroynatis	Nombre de la base de datos utilizada por el sistema.
Base de datos	DB_USERNAME	root	Nombre del administrador de base de datos (predeterminado)

	Gero y Natis Plan de instalación	SENA
---	---	-------------

Servidor SMTP	MAIL_HOST	smtp.gmail.com	Servidor de correo utilizado para enviar notificaciones o formularios.
Roles de usuario	idRol	Administrador = 1 Vendedor = 2	Define los niveles de acceso y permisos en el sistema.

4.4 Personalización

Personalizaciones requeridas:

- Control de acceso por roles (vendedor, administrador), con vistas limitadas según permisos.
- Restricción para que los vendedores solo accedan al módulo de ventas desde la caja. Personalización del diseño visual con Bootstrap para una interfaz responsiva adaptada a la tienda.
- Inclusión de carga de imágenes a través de Cloudinary desde el formulario de productos.
- Validación de formularios personalizada con JavaScript (sin frameworks externos).

Desarrollo adicional:

- Implementación de lógica para enviar correos automáticos (confirmación de registro, contacto) con PHPMailer.
- Integración de Cloudinary mediante Composer para manejo externo de imágenes.
- Creación de un sistema de backup manual a través de scripts SQL exportables desde phpMyAdmin.

Configuración de informes:



- Calendario que filtra las ventas de un día, que facturas hay, el total y el total de las ventas por día.

4.5 Migración de datos

En este proyecto no se requiere una migración de datos automatizada, ya que la información previa estaba registrada en formato físico (cuadernos). Los datos serán ingresados manualmente en el sistema durante su implementación.

5 CRONOGRAMA

5.1 Calendario de actividades

ID	Actividad	Duración	Fecha inicio	Fecha fin	Predecesor	Responsable
A1	Preparación del entorno	1 día	10/06/2025	11/06/2025	-	Vanessa Mateus
A2	Instalación de prerequisites	1 día	10/06/2025	11/06/2025	A1	Daniel Bermeo
A3	Instalación de software base	1 día	10/06/2025	11/06/2025	A2	Deisy González
A4	Configuración inicial	1 día	10/06/2025	11/06/2025	A3	Deisy González
A5	Pruebas de funcionamiento	2 días	11/06/2025	12/06/2025	A4	Daniel Bermeo
A6	Validación de acceso por roles	1 día	12/06/2025	13/06/2025	A5	Vanessa Mateus
A7	Puesta en producción	3 días	13/06/2025	16/06/2025	A6	Daniel Bermeo

5.2 Hitos clave

#	Hito	Fecha objetivo	Entregables
1	Preparación del entorno	11/06/2025	Entorno local con XAMPP y Visual Studio Code listos



2	Instalación de prerrequisitos	11/06/2025	Git, Composer, dependencias instaladas
3	Instalación de software base	11/06/2025	Proyecto clonado y funcionando en navegador local
4	Configuración inicial	11/06/2025	Sistema conectado a base de datos y servicios externos
5	Pruebas de funcionamiento	12/06/2025	Módulos verificados: login, productos, ventas, reportes
6	Validación de acceso por roles	13/06/2025	Accesos funcionales según tipo de usuario (admin/vendedor)
7	Puesta en producción	16/06/2025	Sistema en Azure funcionando con datos reales

6 Pruebas

6.1 Estrategia de pruebas

La estrategia de pruebas del proyecto Gero y Natis se basa en un enfoque incremental, combinando pruebas manuales y automáticas. Se validará cada módulo a medida que se complete, utilizando entornos separados para pruebas y desarrollo. El objetivo es detectar errores de forma temprana, asegurar la calidad del software y garantizar que cumpla con los requisitos funcionales y no funcionales acordados.



6.2 Tipos de pruebas a realizar

- **Pruebas de instalación:**
Validación de que el sistema puede ser instalado correctamente en diferentes entornos (local, staging y producción), incluyendo configuración de dependencias, base de datos y servicios.
- **Pruebas funcionales:**
Se evaluará cada funcionalidad clave del sistema: registro de usuarios, inicio de sesión, asignación de roles, navegación del sistema, entre otros, según los casos de uso definidos.
- **Pruebas de integración:**
Comprobación del correcto funcionamiento de los módulos integrados (login con base de datos, manejo de sesiones, autenticación y relaciones entre usuarios y roles).
- **Pruebas de rendimiento:**
Se simulará carga de múltiples usuarios para validar que el sistema mantiene tiempos de respuesta aceptables y no se satura ante la concurrencia.
- **Pruebas de seguridad:**
Verificación de protección contra accesos no autorizados, validación de sesiones, cifrado de contraseñas y restricción de funciones según roles.
- **Pruebas de aceptación de usuario (UAT):**
Se invitará a usuarios clave (cliente o usuarios finales) a interactuar con el sistema y confirmar que cumple con las expectativas y requerimientos establecidos.

ID	Caso de prueba	Criterio	Responsable
----	----------------	----------	-------------



CP1	Registro e inicio de sesión	El usuario debe poder registrarse e iniciar sesión correctamente con sus credenciales.	Deisy Gonzalez
CP2	Acceso según rol asignado	El sistema debe mostrar solo los módulos permitidos según el rol (admin o vendedor).	Vanessa Mateus
CP3	Gestión de productos	El administrador debe poder registrar, editar e inhabilitar productos.	Daniel Bermeo
CP4	Registro de ventas	El vendedor debe poder registrar una venta desde su interfaz.	Deisy Gonzalez

7 EQUIPO DE RESPONSABILIDADES

7.1 Estructura del equipo

Rol	Nombre	Departamento	Responsabilidad
Líder del proyecto	Deisy González	Gestión de proyectos	Supervisar la ejecución del proyecto y coordinar al equipo.
Desarrollador Back end	Daniel Bermeo	Desarrollo	Implementar la lógica del sistema y conexión con la base de datos.



Desarrollador Front end	Vanessa Mateus	Desarrollo	Diseñar e implementar la interfaz del usuario.
----------------------------	----------------	------------	---

7.2 Matriz RACI

Actividad	Deisy González	Vanessa Mateus	Daniel Bermeo
Definición de requisitos	A	R	C
Desarrollo de funcionalidades	I	C	R
Diseño de interfaz	I	R	C
Configuración de base de datos	I	C	R
Pruebas funcionales	R	C	I
Validación con usuario	R	I	C
Entrega final del sistema	R	C	I



7.3 Escalamiento

Nivel	Tipo de incidente	Contacto	Tiempo de respuesta
1	Falla critica del sistema	Vanessa Mateus	Máximo 1 hora
2	Error funcional importante sin detener el sistema	Daniel Bermeo	Máximo 4 horas hábiles
3	Incidente menor o solicitud de mejora	Deisy González	1-2 días hábiles

8 GESTIÓN DE RIESGOS

8.1 Matriz de riesgos

ID	Riesgo	Probabilidad	Impacto	Criticidad	Estrategia de mitigación	Responsable
R1	Fallos en la base de datos	Alta	Alta	Alta	Respaldo s diarios automáticos y pruebas periódicas de restauración	Deisy González



R2	Retraso en desarrollo de funcionalidades	Media	Alta	Alta	Planificación ágil y control semanal de avances.	Daniel Bermeo
R3	Vulnerabilidades de seguridad en login	Media	Alta	Alta	Uso de hashing seguro y validación de entradas.	Vanessa Mateus
R4	Inestabilidad en interfaz de usuario	Baja	Media	Modorada	Pruebas continuas en frontend antes de despliegue.	Daniel Bermeo



R5	Fallas en pruebas de aceptación del usuario	Baja	Alta	Modera da	Realizar pruebas UAT anticipadas con usuarios clave.	Deisy González

9 CAPACITACIÓN Y SOPORTE

9.1 Plan de capacitación

Grupo de usuarios	Contenido	Modalidad	Duración	Fecha planificada
Administradores	Gestión de usuarios, asignación de roles, reportes		2 horas	21/06/2025



Usuarios finales	Registro inicio de sesión, navegación general		1 hora	22/06/2025
Soporte técnico (QA)	Validación de errores, pruebas, rollback		1.5 horas	23/06/2025

9.2 Soporte post-implementación

Nivel 1 – Soporte básico al usuario final:

Atención a dudas de uso, fallos menores o errores visuales.

Contacto: Vanessa Mateus – vanesamateus@gmail.com

Horario: Lunes a viernes, 9:00 a.m. – 5:00 p.m.

Nivel 2 – Soporte técnico funcional:

Resolución de errores relacionados con funcionalidades, formularios o procesos del sistema.

Contacto: Daniel Bermeo – felipedanieltorres32@gmail.com

Horario: Lunes a viernes, 9:00 a.m. – 5:00 p.m.

Nivel 3 – Soporte de gestión o incidencias críticas:

Cambios estructurales, restauración de copias de seguridad o caída del sistema.

Contacto: Deisy González – deisygonzalezbueno2007@gmail.com



Gero y Natis
Plan de instalación

SENA

Horario: Lunes a viernes, 9:00 a.m. – 6:00 p.m.

Soporte del proveedor:

Al ser un desarrollo propio, no existe un proveedor externo. El soporte es gestionado directamente por el equipo desarrollador del proyecto.