**PROCESO:**

**MANUALES DE IMPLEMENTACION Y DE USO DE BACKEND Y FRONTEND**

**MANUAL:**

**USO DEL BACKEND**

|  |  |  |
| --- | --- | --- |
| **CONTROL DE CAMBIOS** | | |
| **VERSIÓN** | **DESCRIPCIÓN DE LA MODIFICACIÓN** | **FECHA** |
| 0 | Emisión inicial | 23/07/2025 |
|  |  |  |
|  |  |  |
|  |  |  |

**CONTENIDO**

[INTRODUCCIÓN 4](#_Toc204063529)

[1 OBJETIVO 4](#_Toc204063530)

[2 ALCANCE 4](#_Toc204063531)

[3 REFERENCIA NORMATIVA 4](#_Toc204063532)

[4 DEFINICIONES 4](#_Toc204063533)

[5 RESPONSABILIDADES 4](#_Toc204063534)

[6 DESARROLLO DEL MANUAL 4](#_Toc204063535)

[10.1.1 📌 ¿Qué es la imagen campanas\_api.png? 5](#_Toc204063536)

[10.1.2 📌 ¿Qué significa un GET en Python/FastAPI? 5](#_Toc204063537)

[10.1.3 📌 ¿Qué es fastapi.middleware.cors? 6](#_Toc204063538)

[13 ANEXOS Y/O DOCUMENTO ASOCIADOS. 13](#_Toc204063539)

# INTRODUCCIÓN

Este manual describe detalladamente el procedimiento para la correcta instalación, configuración, ejecución y verificación del backend del proyecto Gateway Nexum, implementado en Python utilizando el framework FastAPI. El documento incluye la instalación de dependencias, configuración básica de rutas, comando para poner en marcha el servidor y las URLs para comprobar su funcionamiento.

# OBJETIVO

Proveer una guía clara y completa que permita a desarrolladores, testers y demás personal técnico instalar, configurar y ejecutar el backend del Gateway Nexum de manera correcta y segura.

# ALCANCE

Este manual aplica a todos los miembros del equipo técnico encargados de la validación, mantenimiento y despliegue del backend de la API Gateway Nexum en entornos de desarrollo o pruebas.

# REFERENCIA NORMATIVA

No aplica

# DEFINICIONES

**• Backend: Componente servidor que gestiona las reglas de negocio y responde a las solicitudes provenientes del cliente o frontend. • API: Conjunto de rutas que permiten la comunicación entre diferentes sistemas de software. • Postman: Herramienta para pruebas y desarrollo de APIs, utilizada aquí para verificar el correcto funcionamiento del backend.**

# RESPONSABILIDADES

• Analista de Pruebas: Ejecutar las instrucciones contenidas en este manual para verificar el correcto funcionamiento del backend y documentar los resultados. • Desarrollador Backend: Garantizar que la API esté actualizada, operativa y configurada según las especificaciones técnicas.

# DESARROLLO DEL MANUAL

1. **Instalación previa**
2. Asegúrese de tener instalado Python 3.11 o superior en su equipo.
3. Ubíquese en la carpeta del proyecto backend en una terminal.
4. Ejecute el siguiente comando para instalar las dependencias necesarias:



Esto descargará las librerías y módulos requeridos para ejecutar la aplicación.

1. **Configuración (rutas)**

* Verifique si existe un archivo .env o archivos de configuración de rutas según las instrucciones del equipo de desarrollo.
* Por defecto, el backend escucha en <http://127.0.0.1:8000>.

Y la url de la pagina seria: <http://172.18.73.76:3000>.

Ingresas el usuario y la clave que te corresponda y podrás observar todos los módulos que se encuentran disponibles con todos los datos.

1. **Cómo correr el servidor**
2. Desde la terminal, ubicado en la carpeta del proyecto, ejecute:



1. El servidor arrancará y quedará escuchando en el puerto por defecto 8000.
2. **URLs de verificación**

Una vez iniciado el servidor, puede verificar su funcionamiento accediendo a las siguientes URLs en su navegador:

* Documentación automática (Swagger UI):



* Documentación alternativa (ReDoc):
* Endpoints de la API (/gateway/...) accesibles desde Postman o navegador.

### 📌 ¿Qué es la imagen campanas\_api.png?

Es una captura tomada en Postman que muestra una prueba al endpoint /api/campanas con el método GET. En la imagen se ve la solicitud enviada y la respuesta JSON obtenida, sirviendo como referencia visual para pruebas exitosas.

### 📌 ¿Qué significa un GET en Python/FastAPI?

El método GET es una solicitud HTTP para obtener datos de un servidor. En FastAPI, las rutas (@app.get(...)) están programadas para responder a estas solicitudes devolviendo datos en formato JSON. Cuando haces un GET desde Postman, FastAPI ejecuta la lógica definida para esa ruta.

### 📌 ¿Qué es fastapi.middleware.cors?

Es un módulo incluido en FastAPI para configurar CORS (Cross-Origin Resource Sharing). Permite aceptar solicitudes desde otros dominios distintos al servidor. En las pruebas con Postman no es necesario configurarlo porque Postman no aplica las restricciones de CORS.

1. **🔷 Módulos principales del backend**

**RPA RUES**

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

**AREAS**

**Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.**

**VIGILANCIA**

**Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.**

**SUPER NOTARIADO**

**Interfaz de usuario gráfica, Tabla

El contenido generado por IA puede ser incorrecto.**

**PERMISOS POR USUARIO**

Captura de pantalla de computadora

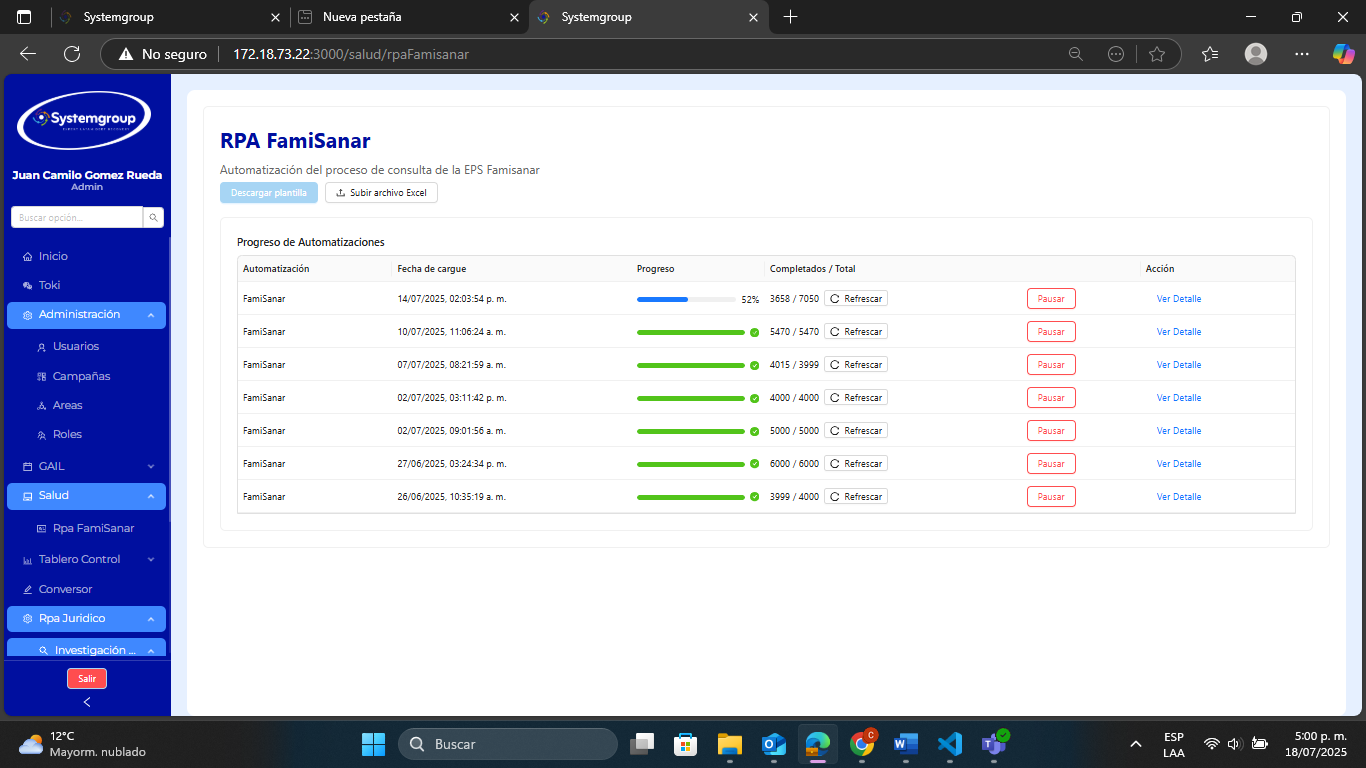
El contenido generado por IA puede ser incorrecto.

**PERMISOS POR ROL**

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

**SALUD**

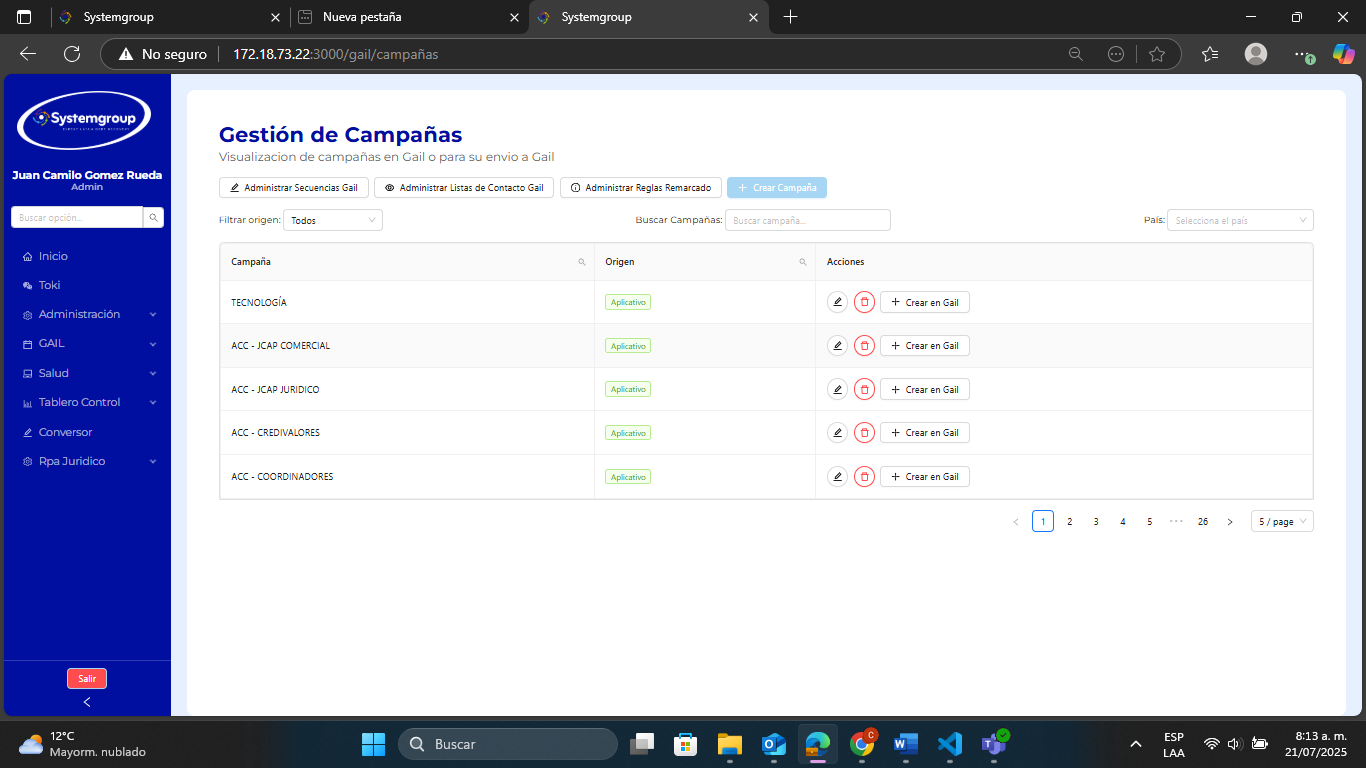


**RUNT**

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

**CAMPAÑAS GAIL**

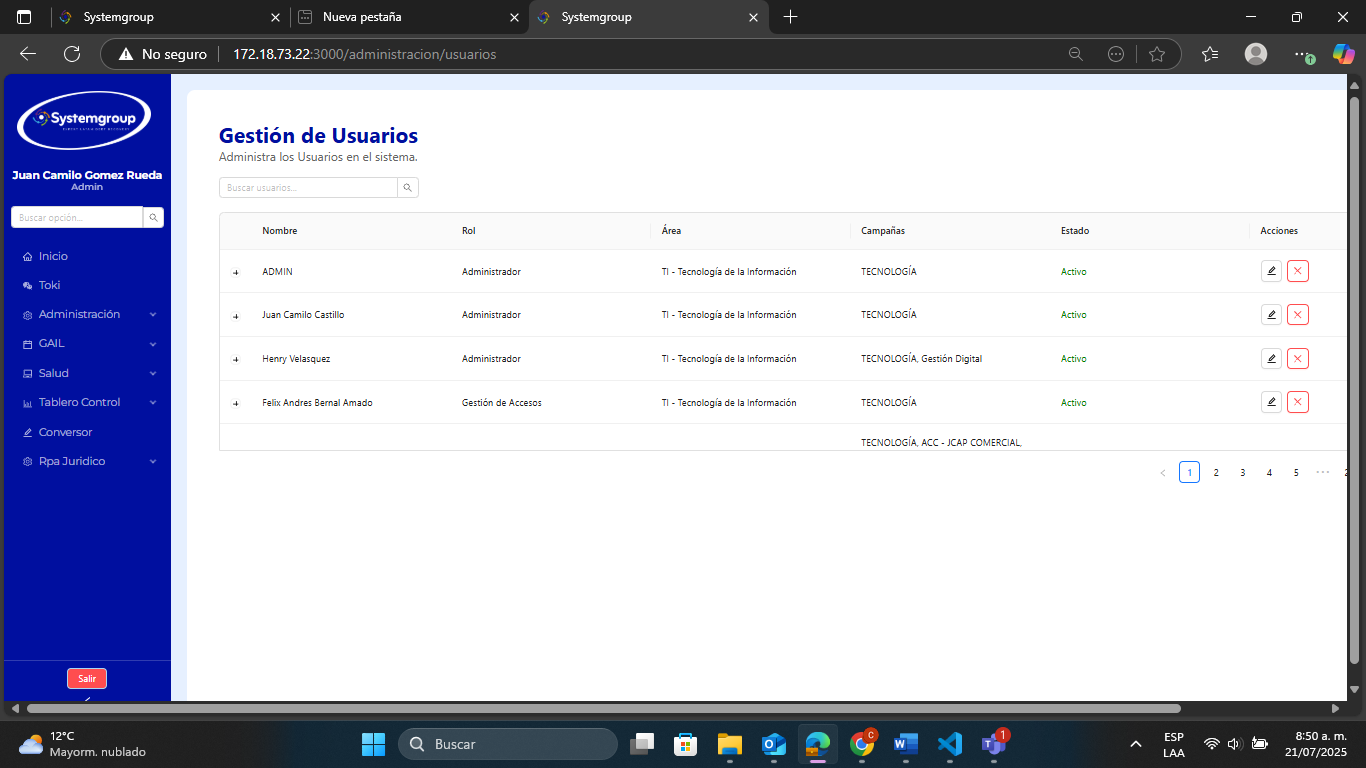


**SALUD (RPA FamiSanar)**

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

**USUARIOS**



**ROLES Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.**

El backend expone los siguientes módulos funcionales a través de endpoints:

| **Módulo** | **Endpoint** | **Método** | **Descripción** |
| --- | --- | --- | --- |
| **Permisos por Usuario** | /gateway/permisos/usuario | GET | Devuelve los permisos asignados al usuario autenticado. |
| **Permisos por Rol** | /gateway/permisos/rol | GET | Devuelve los permisos asociados a un rol específico. |
| **RUNT** | /gateway/runt | GET | Consulta la información registrada en el sistema RUNT. |
| **SALUD** | /gateway/salud | GET | Este módulo permite a los usuarios automatizar las consultas a la EPS Famisanar, facilitando el procesamiento de datos relacionados con afiliaciones o estado de los usuarios en esa EPS. |
| **CAMPAÑAS** | /gateway/campanas | GET | Permite a los usuarios crear, administrar y dar seguimiento a las campañas activas dentro de la organización. Desde este módulo, se pueden ver estadísticas, editar campañas existentes y crear nuevas campañas según las necesidades del negocio. |
| **AREAS** | /gateway/areas | GET | Este módulo permite gestionar las diferentes áreas u oficinas dentro de la organización. Los usuarios con permisos pueden crear nuevas áreas, modificar las existentes y asignar responsabilidades a cada una. |
| **VIGILANCIA** | /gateway/vigilancia | GET | El módulo de Vigilancia brinda herramientas para el monitoreo y auditoría de procesos, asegurando el cumplimiento de los estándares  internos y detectando irregularidades en las operaciones. |
| **SUPERNOTARIADO** | /gateway/supernotariado | GET | Facilita las consultas y trámites relacionados con el sistema notarial. Los usuarios pueden automatizar consultas a registros notariales y obtener la información requerida directamente desde la plataforma. |
| **RPA RUES** | /gateway/rpa\_rues | GET | Este módulo automatiza los trámites relacionados con el Registro Único Empresarial y Social (RUES), ahorrando tiempo y reduciendo errores al realizar estas gestiones manualmente. |
| **CAMPAÑAS GAIL** | |  | | --- | |  |  |  | | --- | | /gateway/campanas/gail | | GET | Similar al módulo Campañas, pero especializado para las campañas gestionadas en el marco del grupo GAIL. Incluye funcionalidades específicas para estas campañas y estadísticas enfocadas en su seguimiento. |
| **Salud (RPA FamiSanar)** | |  | | --- | |  |  |  | | --- | | /gateway/salud/famisanar | | GET | Automatiza las consultas a la EPS Famisanar, permitiendo a los usuarios subir un archivo Excel con datos y obtener respuestas sobre el estado de afiliación o consultas médicas. |
| **USUARIOS Y ROLES** | /gateway/permisos/usuario  /gateway/permisos/rol | GET | Parte del módulo Administración. Permite gestionar usuarios registrados en el sistema, asignarles roles y definir sus permisos de acceso. |

Para probar cada módulo, use Postman con el método GET y el endpoint correspondiente.

1. **NOTAS ADICIONALES**

* CORS ya está habilitado para desarrollo, no requiere cambios para pruebas en Postman.
* Si la respuesta no es la esperada, asegúrese de que el backend esté en ejecución y la URL sea correcta.

# ANEXOS Y/O DOCUMENTO ASOCIADOS.

|  |  |
| --- | --- |
| **CÓDIGO** | **NOMBRE DEL DOCUMENTO** |
| No Aplica | No Aplica |

|  |  |  |
| --- | --- | --- |
| **ELABORÓ** | **REVISÓ** | **APROBÓ** |
| **Nombre:** Juan Camilo Gomez Rueda  **Cargo:** Aprendiz Sena | **Nombre:** Henry Velásquez  **Cargo:** Director de Riesgos Tecnológicos y Continuidad del Negocio | **Nombre:** Jesika Rojas  **Cargo:** Gerente de Riesgos y Oficial de Cumplimiento |
| **Fecha: 23/07/2025** | **Fecha:**  **23/02/2023** | **Fecha:**  **24/02/2023** |