



Sistema de Gestión Policlínico

Manual de Usuario

Versión 1.0

15 de diciembre de 2025

Nombre	Telegram
Diego Hernández Rodríguez	@eu_sou_dieguinho
Sammy Raul Sosa Justiz	@Sosa_Samy
Daniela De La Caridad Guerrero Álvarez	@DGA47
Rubén Martínez Rojas	@Nebur02

Índice

1. Introducción	3
1.1. Propósito del Manual	3
1.2. Descripción del Sistema	3
1.3. Objetivos del Proyecto	3
2. Instalación y Configuración	3
2.1. Requisitos del Sistema	3
2.1.1. Requisitos Mínimos de Hardware	3
2.1.2. Requisitos Recomendados	4
2.2. Requisitos de Software	4
2.3. Instalación Paso a Paso	4
2.3.1. Paso 1: Instalar Dependencias	4
2.3.2. Paso 2: Configuración del Proyecto	5
2.3.3. Paso 3: Verificación de la Instalación	5
2.4. Configuración de Base de Datos	6
2.4.1. Credenciales de pgAdmin	6
2.4.2. Conexión a PostgreSQL	6
3. Guía del Usuario	7
3.1. Acceso al Sistema	7
3.1.1. Pantalla de Inicio de Sesión	7
3.2. Panel Principal (Dashboard)	8
3.2.1. Elementos del Dashboard	8
3.3. Módulos del Sistema	8
3.3.1. Módulo de Trabajadores	8
3.3.2. Módulo de Departamentos	8
3.3.3. Módulo de Pacientes	8
3.3.4. Módulo de Atenciones Médicas	8
3.3.5. Módulo de Medicamentos	9
3.3.6. Historial Clínico Electrónico (HCE)	9
3.3.7. Panel de Control de Stock	10
3.3.8. Módulo de Reportes	10
3.3.9. Módulo de Administración	11
4. Operaciones Avanzadas	11
4.1. Comandos Útiles	11
4.1.1. Gestión de Docker	11
4.1.2. Gestión de la Aplicación	11
4.2. Solución de Problemas	12
4.2.1. Problemas Comunes	12
4.2.2. Verificación de Puertos	12
5. Referencia Técnica	12
5.1. Estructura del Proyecto	12
5.2. Configuración de Docker Compose	12
5.3. Variables de Entorno	13

6. Soporte y Mantenimiento	13
6.1. Canales de Soporte	13
6.2. Mantenimiento Preventivo	14
6.2.1. Copias de Seguridad	14
6.2.2. Actualizaciones	14
6.2.3. Monitorización	14
6.3. Consideraciones de Seguridad	14
7. Anexos	14
7.1. Script de Instalación Rápida	14
7.2. Puertos del Sistema	15
7.3. Consideraciones Finales	15

1 Introducción

1.1 Propósito del Manual

Este manual proporciona instrucciones completas para la instalación, configuración y uso del Sistema de Gestión Policlínico "Policlinic". Está dirigido a administradores del sistema, personal técnico y usuarios finales.

1.2 Descripción del Sistema

Policlinic es una aplicación web integral diseñada para centralizar y automatizar la gestión de procesos médicos y administrativos en policlínicos docentes. El sistema ofrece funcionalidades completas para la gestión de pacientes, personal médico, inventario de medicamentos y generación de reportes.

1.3 Objetivos del Proyecto

- **Objetivo General:** Automatizar y centralizar la gestión de procesos médicos y administrativos en policlínicos.
- **Objetivos Específicos:**
 - Gestión integral del personal médico y administrativo
 - Mantenimiento de historial clínico electrónico centralizado
 - Registro de atenciones médicas programadas y de urgencia
 - Control de inventario de medicamentos y suministros
 - Generación automatizada de reportes e indicadores de gestión
 - Mejora de la eficiencia operativa y calidad de atención

2 Instalación y Configuración

2.1 Requisitos del Sistema

2.1.1. Requisitos Mínimos de Hardware

Componente	Especificación
Sistema Operativo	Windows 10, macOS 10.14+, Ubuntu 18.04+
Procesador	CPU 64-bit, 2 núcleos
Memoria RAM	4 GB mínimo
Almacenamiento	2 GB espacio libre
Resolución	1280x720 píxeles
Conexión a Internet	Para descarga de dependencias

Cuadro 1: Requisitos mínimos del sistema

2.1.2. Requisitos Recomendados

Componente	Especificación
Sistema Operativo	Última versión estable
Procesador	CPU 64-bit, 4 núcleos
Memoria RAM	8 GB o más
Almacenamiento	5 GB espacio libre
Resolución	1920x1080 píxeles

Cuadro 2: Requisitos recomendados para óptimo rendimiento

2.2 Requisitos de Software

- **Docker y Docker Compose**
 - Versión mínima: Docker 20.10+, Docker Compose 2.0+
 - Descarga: <https://www.docker.com/products/docker-desktop>
- **Node.js**
 - Versión: 20 LTS o superior
 - Recomendado: Instalar via NVM (Node Version Manager)
- **NestJS CLI**
 - Instalación global requerida: `npm install -g @nestjs/cli`

2.3 Instalación Paso a Paso

2.3.1. Paso 1: Instalar Dependencias

Windows:

1. Descargar e instalar Docker Desktop desde el sitio oficial
2. Instalar NVM-Windows:

```

1 # Descargar de: https://github.com/coreybutler/nvm-windows
2 nvm install 20
3 nvm use 20

```

macOS:

```

1 # Instalar Homebrew
2 /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/
   Homebrew/install/HEAD/install.sh)"
3
4 # Instalar Docker y Node.js
5 brew install --cask docker
6 brew install nvm
7 nvm install 20
8 nvm use 20

```

Linux (Ubuntu/Debian):

```

1 # Actualizar paquetes
2 sudo apt update
3
4 # Instalar Docker
5 sudo apt install docker-ce docker-ce-cli containerd.io

```

```

6
7 # Agregar usuario al grupo docker
8 sudo usermod -aG docker $USER
9
10 # Instalar Docker Compose
11 sudo curl -L "https://github.com/docker/compose/releases/download/
12   /v2.20.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/
13     bin/docker-compose
14 sudo chmod +x /usr/local/bin/docker-compose
15
16 # Instalar NVM y Node.js
17 curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/
18   install.sh | bash
source ~/.bashrc
nvm install 20
nvm use 20

```

Listing 1: Instalación en Linux

2.3.2. Paso 2: Configuración del Proyecto

Backend:

```

1 # Navegar a la carpeta del backend
2 cd backend
3
4 # Instalar dependencias
5 npm install
6
7 # Iniciar contenedores Docker
8 docker compose up -d
9
10 # Iniciar servidor de desarrollo
11 npm run start:dev

```

Listing 2: Configuración del Backend

Frontend:

```

1 # En nueva terminal, navegar al frontend
2 cd ../frontend
3
4 # Instalar dependencias
5 npm install --legacy-peer-deps
6
7 # Iniciar servidor de desarrollo
8 npm run dev

```

Listing 3: Configuración del Frontend

2.3.3. Paso 3: Verificación de la Instalación

1. Verificar contenedores Docker activos:

```
1 docker ps
```

Deberían aparecer 3 contenedores activos.

2. Acceder a las aplicaciones:

- **Frontend:** <http://localhost:5173>
- **Backend (API):** <http://localhost:3000>
- **pgAdmin:** <http://localhost:5050>

2.4 Configuración de Base de Datos

2.4.1. Credenciales de pgAdmin

Parámetro	Valor
URL	http://localhost:5050
Email	admin@policlinic.com
Contraseña	admin123

Cuadro 3: Credenciales de acceso a pgAdmin

2.4.2. Conexión a PostgreSQL

Campo	Valor
Nombre de conexión	PoliclinicDB
Host	policlinic_postgres
Puerto	5432
Base de datos	poldb
Usuario	policlinic
Contraseña	policlinicpass

Cuadro 4: Configuración de conexión a PostgreSQL

3 Guía del Usuario

3.1 Acceso al Sistema

3.1.1. Pantalla de Inicio de Sesión



Figura 1: Pantalla de inicio de sesión

Campos de acceso:

- **Usuario:** Ingresar nombre de usuario o email institucional
- **Contraseña:** Contraseña personal (se muestra oculta)
- **Botón "Ingresar":** Validar credenciales y acceder al sistema

3.2 Panel Principal (Dashboard)

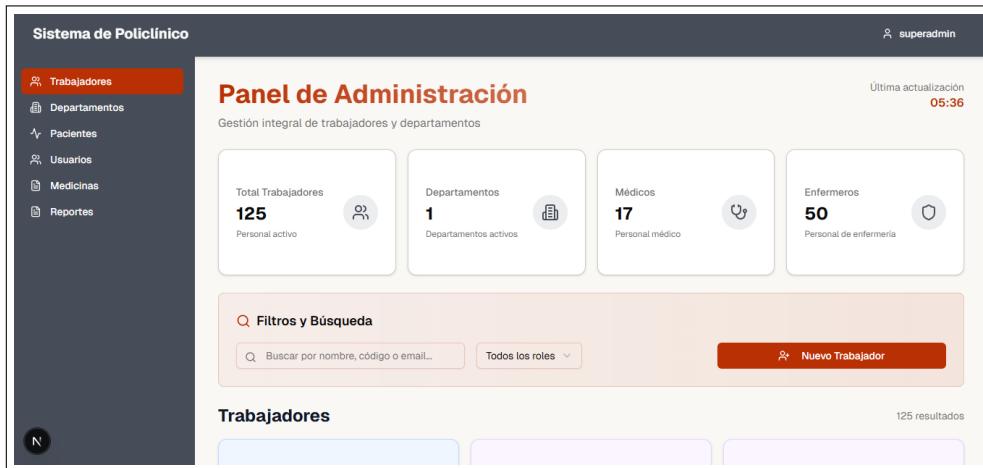


Figura 2: Panel principal del sistema

3.2.1. Elementos del Dashboard

- **Barra de Navegación Superior:**
 - Logo del sistema
 - Perfil de usuario activo
 - Opción para cerrar sesión
- **Menú Lateral:** Acceso a todos los módulos del sistema
- **Área de Trabajo Principal:** Contenido específico del módulo seleccionado

3.3 Módulos del Sistema

3.3.1. Módulo de Trabajadores

- **Registrar Personal:** Formulario para nuevo personal médico/administrativo
- **Listado de Personal:** Tabla con filtros por departamento, cargo y estado
- **Asignar Departamento:** Asignar personal a unidades organizativas
- **Designar Jefaturas:** Asignar responsables de departamento
- **Actualizar Estado:** Cambiar estado laboral (activo, licencia, baja)

3.3.2. Módulo de Departamentos

- **Nuevo Departamento:** Crear nuevas unidades organizativas
- **Gestión de Departamentos:** Editar información, asignar personal y designar jefes

3.3.3. Módulo de Pacientes

- **Nuevo Paciente:** Registro completo de datos personales
- **Consultar Pacientes:** Búsqueda por nombre, ID, edad, etc.
- **Actualizar Datos:** Modificar información del paciente

3.3.4. Módulo de Atenciones Médicas

- **Nueva Consulta Programada:** Programar cita con remisión/derivación

- **Registro de Urgencia:** Atención inmediata en cuerpo de guardia
- **Listado de Atenciones:** Historial de atenciones médicas
- **Prescripción Médica:** Formulario para prescripciones de medicamentos

3.3.5. Módulo de Medicamentos

- **Catálogo de Medicamentos:** Administración de todos los medicamentos disponibles
- **Control de Stock:** Gestión de inventario por departamento

3.3.6. Historial Clínico Electrónico (HCE)

Vista del Historial Clínico

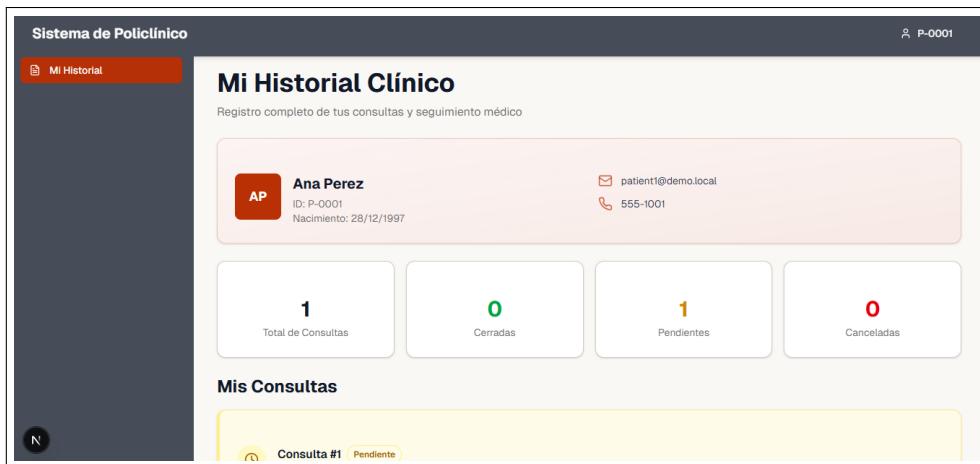


Figura 3: Interfaz del Historial Clínico Electrónico

Secciones del HCE:

- **Datos Personales:** Información básica del paciente
- **Atenciones Cronológicas:** Listado ordenado por fecha
- **Diagnósticos:** Enfermedades y condiciones registradas
- **Tratamientos:** Medicamentos y terapias prescritas

3.3.7. Panel de Control de Stock

Stock de Departamento

The screenshot shows the 'Stock del Departamento' section of the system. At the top, there are four summary boxes: 'Total Medicamentos' (1 tipo diferente), 'Disponibles' (1 Stock normal), 'Stock Bajo' (0 Requerien atención), and 'Unidades Totales' (76 Cantidad total). Below this is a search bar with a placeholder 'Buscar por nombre o código...' and a dropdown menu 'Todos los estados'. A green header bar indicates '1 resultados'.

Figura 4: Panel de control de inventario

Funcionalidades:

- **Vista por Departamento:** Stock específico de cada área
- **Alertas Automáticas:** Notificaciones de niveles bajos/críticos
- **Registro de Movimientos:** Entradas y salidas del inventario
- **Solicitud de Reposición:** Generar pedidos al almacén central
- **Recepción de Mercancía:** Confirmación de productos recibidos

3.3.8. Módulo de Reportes

Generador de reportes

The screenshot shows the 'Reportes' section. It features a grid of report types: Consultas, Medicamentos, Remisiones, Pacientes, Personal, and Departamentos. Below this are filter fields for 'Departamento' (Todos), 'Fecha Inicio' (mm/dd/yyyy), 'Fecha Fin' (mm/dd/yyyy), and 'Estado' (Todos). A 'Generar Reporte' button is at the bottom right.

Figura 5: Interfaz para generación de reportes

Tipos de reportes disponibles:

- **Consultas:** Reportes de atenciones médicas
- **Pacientes:** Estadísticas de pacientes atendidos
- **Remisiones:** Informes de derivaciones médicas

- **Medicamentos:** Consumo y stock de fármacos
- **Personal:** Información del personal médico y administrativo
- **Departamentos:** Actividad por unidad organizativa
 - Características:**
 - Filtros avanzados por fechas, departamentos y tipos
 - Exportación a PDF con un clic
 - Visualización previa antes de exportar

3.3.9. Módulo de Administración

- **Gestión de Usuarios:** Crear, modificar y desactivar usuarios
- **Configuración del Sistema:** Parámetros generales de la aplicación
- **Logs de Auditoría:** Registro de accesos y modificaciones

4 Operaciones Avanzadas

4.1 Comandos Útiles

4.1.1. Gestión de Docker

```

1 # Iniciar todos los servicios
2 docker compose up -d
3
4 # Detener todos los servicios
5 docker compose down
6
7 # Ver logs de un contenedor
8 docker logs policlinic_postgres
9
10 # Ver estado de todos los contenedores
11 docker ps -a
12
13 # Acceder a un contenedor (interactivo)
14 docker exec -it policlinic_postgres bash

```

Listing 4: Comandos Docker esenciales

4.1.2. Gestión de la Aplicación

```

# Backend - Modo desarrollo
npm run start:dev

# Backend - Compilación para producción
npm run build
npm run start:prod

# Frontend - Modo desarrollo
npm run dev

# Frontend - Compilación para producción

```

```
npm run build
```

4.2 Solución de Problemas

4.2.1. Problemas Comunes

Problema	Solución
Docker no inicia	Verificar virtualización en BIOS, reiniciar Docker
Puertos ocupados	Cambiar puertos en docker-compose.yml
Error en npm install	Ejecutar: <code>npm cache clean --force</code>
Contenedores no se crean	Ejecutar: <code>docker compose up --build -d</code>
Acceso denegado	Ejecutar como administrador/sudo
Imágenes no se cargan	Verificar nombres y rutas de archivos

Cuadro 5: Solución de problemas comunes

4.2.2. Verificación de Puertos

```

1 # Windows
2 netstat -ano | findstr :3000
3
4 # Linux/Mac
5 lsof -i :3000      # Backend API
6 lsof -i :5436      # PostgreSQL
7 lsof -i :5173      # Frontend
8 lsof -i :5050      # pgAdmin

```

Listing 5: Verificar puertos ocupados

5 Referencia Técnica

5.1 Estructura del Proyecto

```
policlinico-gestion/
├── backend/
│   ├── src/          # Código fuente del backend
│   ├── docker-compose.yml # Configuración de contenedores
│   ├── package.json    # Dependencias Node.js
│   └── .env           # Variables de entorno
└── frontend/
    ├── src/          # Código fuente del frontend
    ├── public/        # Archivos estáticos
    └── package.json    # Dependencias React
 README.md           # Documentación principal
```

5.2 Configuración de Docker Compose

```
version: '3.8'
services:
  postgres:
```

```

image: postgres:15
container_name: policlinic_postgres
environment:
  POSTGRES_USER: policlinic
  POSTGRES_PASSWORD: policlinicpass
  POSTGRES_DB: poldb
ports:
  - "5436:5432"

mongo:
  image: mongo:7
  container_name: policlinic_mongo

pgadmin:
  image: dpage/pgadmin4:8
  container_name: policlinic_pgadmin
  environment:
    PGADMIN_DEFAULT_EMAIL: admin@policlinic.com
    PGADMIN_DEFAULT_PASSWORD: admin123
  ports:
    - "5050:80"

```

5.3 Variables de Entorno

```

# Base de datos PostgreSQL
DB_HOST=policlinic_postgres
DB_PORT=5432
DB_USERNAME=policlinic
DB_PASSWORD=policlinicpass
DB_DATABASE=poldb

# MongoDB
MONGO_URI=mongodb://policlinic_mongo:27017/policlinic

# Seguridad JWT
JWT_SECRET=tu_secreto_aqui_32_caracteres_minimo
JWT_EXPIRATION=24h

# Configuración de servidor
PORT=3000
NODE_ENV=development

```

6 Soporte y Mantenimiento

6.1 Canales de Soporte

■ Equipo de Desarrollo:

- Diego Hernández Rodríguez: @eu_sou_dieguinho
- Sammy Raul Sosa Justiz: @Sosa_Samy
- Daniela De La Caridad Guerrero Álvarez: @DGA47

- Rubén Martínez Rojas: @Nebur02
- **Repositorios del Proyecto:**
 - Frontend: <https://github.com/SammyRSosa/nest-p-fe.git>
 - Backend: <https://github.com/SammyRSosa/policlinic-backend.git>
- **Reporte de Problemas:** A través de issues en GitHub

6.2 Mantenimiento Preventivo

6.2.1. Copias de Seguridad

- Realizar backup diario de las bases de datos
- Guardar copias en ubicación externa segura
- Verificar integridad de backups regularmente

6.2.2. Actualizaciones

- Mantener Docker y Node.js en versiones soportadas
- Actualizar dependencias de seguridad periódicamente
- Probar actualizaciones en entorno de desarrollo antes de producción

6.2.3. Monitorización

- Verificar espacio en disco regularmente
- Monitorear uso de CPU y memoria
- Revisar logs de aplicación y base de datos

6.3 Consideraciones de Seguridad

- Cambiar contraseñas por defecto después de la instalación
- Implementar firewall para puertos expuestos
- Usar certificados SSL/TLS en producción
- Restringir acceso físico a servidores
- Implementar políticas de contraseñas seguras

7 Anexos

7.1 Script de Instalación Rápida

```
1 #!/bin/bash
2 # Script de instalacion para Linux
3
4 echo "==== Instalacion Sistema Gestion Policlinico ===="
5
6 # Instalar Docker
7 echo "Instalando Docker..."
8 sudo apt update
9 sudo apt install docker.io docker-compose
10
11 # Instalar Node.js
12 echo "Instalando Node.js..."
13 curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
14 sudo apt install nodejs
```

```

15
16 # Clonar repositorio
17 echo "Clonando repositorio..."
18 git clone [URL_REPO]
19 cd policlinico-gestion
20
21 # Iniciar servicios
22 echo "Iniciando servicios..."
23 cd backend
24 npm install
25 docker-compose up -d
26 npm run start:dev &
27
28 cd ../frontend
29 npm install --legacy-peer-deps
30 npm run dev &
31
32 echo "==== Instalacion completada ==="
33 echo "Frontend: http://localhost:5173"
34 echo "Backend: http://localhost:3000"
35 echo "pgAdmin: http://localhost:5050"

```

Listing 6: Script de instalación automatizada

7.2 Puertos del Sistema

Servicio	Puerto	Descripción
Frontend React	5173	Interfaz de usuario principal
Backend API	3000	Servicio REST API
PostgreSQL	5436	Base de datos principal
pgAdmin	5050	Administración de PostgreSQL
MongoDB	27017	Base de datos NoSQL

Cuadro 6: Puertos utilizados por el sistema

7.3 Consideraciones Finales

- **Reinicio necesario:** Despues de instalar Docker, reinicie el equipo
- **Permisos Linux:** Configure permisos para ejecutar Docker sin sudo
- **Actualizaciones:** Mantenga software actualizado a versiones recomendadas
- **Espacio en disco:** Verifique espacio disponible antes de actualizaciones
- **Backup:** Realice backup completo antes de cambios importantes
- **Documentación:** Consulte repositorios para actualizaciones del manual