



MANUAL TÉCNICO DE GESDOC & TRAIN

Johan Verjan Carillo & Carlos Culma



28 DE NOVIEMBRE DE 2025
CORPORACIÓN UNIVERSITARIA DEL HUILA
VERSIÓN 1

Contenido

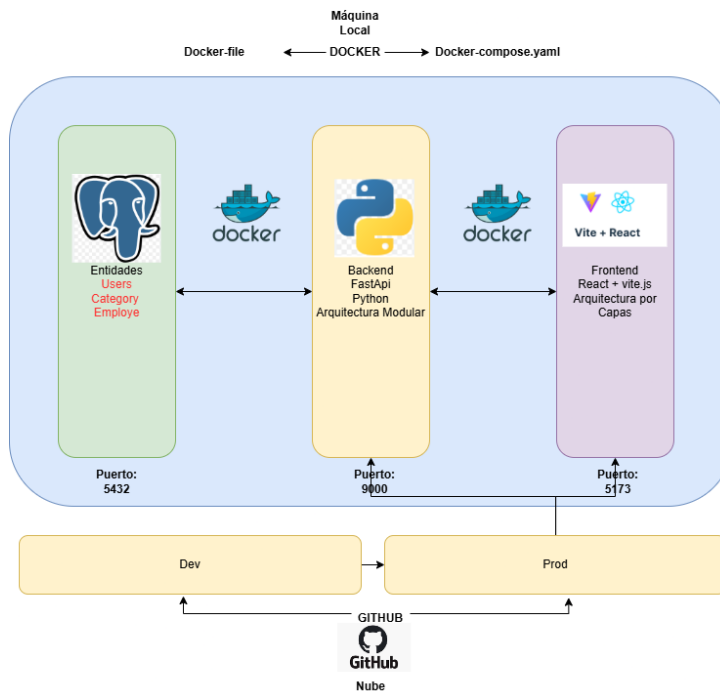
1. Descripción General del Sistema.....	2
2. Arquitectura del Sistema.....	2
2.1 Frontend: (React)	3
2.2 Backend (FastAPI – Python versión 11-14).....	3
2.3. Base de datos (PostgreSQL)	3
2.4. Dockerización	3
3. Instalación del sistema	4
3.1. Requisitos.....	4
3.2. Clonar repositorio	4
4. Backend – Instalación	4
4.1. Crear entorno virtual.....	4
4.2. Instalar dependencias	4
4.3. Ejecutar backend	4
5. Frontend – Instalación.....	5
5.1. Instalar paquetes	5
5.2. Ejecutar frontend.....	5
6. Base de Datos – Migraciones	5
6.1 Crear tablas:	5
7. Docker – Puesta en marcha	5
7.1 Con todos los servicios configurados:.....	5
7.2 Servicios accesibles:	5
8. Modelo de Datos	6
9. Endpoints principales del Backend (FastAPI)	6
10. Módulo de Seguridad	7
10.1 Incluye:.....	7
10.2 Roles + Permisos	7
11. Logs y Auditoría	7
11.1 Se registra:	7

1. Descripción General del Sistema

GESDOC & TRAIN es una aplicación web diseñada para la gestión documental y el control de entrenamientos internos. Integra los procesos de:

- Gestión de roles (Security)
- Gestión de usuarios
- Gestión de documentos
- Gestión de entrenamientos y asistencias
- Auditoría y trazabilidad de acciones
- Seguridad y autenticación JWT

2. Arquitectura del Sistema



La arquitectura está distribuida en tres capas principales:

2.1 Frontend: (React)

- React versión 18
- React Router
- JWT Local Storage
- Fetch API / Axios
- Diseño adaptativo (CSS, Tailwind opcional)

2.2 Backend (FastAPI – Python versión 11-14)

- FastAPI versión 0.109
- SQLAlchemy 2.0
- Autenticación JWT
- Router modular /auth, /user, /role, /documents, /training

2.3. Base de datos (PostgreSQL)

Tablas principales:

- role
- user
- role_assignment
- document_category
- document
- training
- training_attendance

2.4. Dockerización

Servicios:

- frontend
- backend
- postgres
- pgadmin

3. Instalación del sistema

3.1. Requisitos

Componente	Versión
Python	3.11-3.14
Node.js	18+
PostgreSQL	14+
Docker Desktop	Última versión
Git	Última versión

3.2. Clonar repositorio

```
git clone https://github.com/tu-org/repo-gesdoc-train.git
```

```
cd repo-gesdoc-train
```

4. Backend – Instalación

4.1. Crear entorno virtual

```
python -m venv env
```

```
source env/bin/activate # Linux
```

```
env\Scripts\activate # Windows Modo de desarrollo
```

4.2. Instalar dependencias

```
pip install -r requirements.txt
```

4.3. Ejecutar backend

```
uvicorn main:app --reload --port 9000
```

5. Frontend – Instalación

5.1. Instalar paquetes

`npm install`

5.2. Ejecutar frontend

`npm start`

6. Base de Datos – Migraciones

6.1 Crear tablas:

`psql -U postgres -f init.sql`

7. Docker – Puesta en marcha

7.1 Con todos los servicios configurados:

`docker compose up --build`

7.2 Servicios accesibles:

- Frontend → <http://localhost:3000>
- Backend → <http://localhost:8000/docs>
- PostgreSQL → localhost:5432
- PgAdmin → <http://localhost:5050>

8. Modelo de Datos



Incluye las entidades con cardinalidades 1:M y campos de auditoría.

9. Endpoints principales del Backend (FastAPI)

/auth/:

- POST /login
- POST /register

/user/

- GET /all
- POST /create
- PUT /update/{id}
- DELETE /delete/{id}

/documents/

- GET /list

- POST /upload
- PUT /replace/{id}

/training/

- POST /create
- POST /attendance

10. Módulo de Seguridad

10.1 Incluye:

- JWT Access Token
- Middleware de validación

10.2 Roles + Permisos

- Auditoría

11. Logs y Auditoría

11.1 Se registra:

- Usuario que ejecuta
- Acción (create, update, delete)
- IP
- Fecha – hora
- Entidad afectada