# **Setup rápido (local)**

## **1) Clonar y ubicarse**

git clone https://github.com/Jona238/CapstoneServigenman.git

cd "CapstoneServigenman/FASE 2/Evidencias de Proyecto"

## **2) Backend (Django + DRF)**

Requisitos: Python 3.10/3.11/3.12

cd backend

# Crear/activar venv

# Windows (PowerShell):

python -m venv .venv

.\.venv\Scripts\Activate.ps1

# macOS/Linux:

# python -m venv .venv

# source .venv/bin/activate

# Instalar dependencias (dev usa SQLite, no necesitas Postgres local)

pip install django djangorestframework psycopg2-binary python-dotenv django-cors-headers djangorestframework-simplejwt

# Migraciones y superusuario

python manage.py makemigrations

python manage.py migrate

python manage.py createsuperuser

# Levantar API

python manage.py runserver 8000

URLs:

* API: http://127.0.0.1:8000/api/
* Admin: http://127.0.0.1:8000/admin/
* JWT: POST http://127.0.0.1:8000/api/auth/token/ (JSON: {"username":"...","password":"..."})

Si PowerShell bloquea el venv:  
 Set-ExecutionPolicy -Scope CurrentUser RemoteSigned

## **3) Frontend (Next.js)**

Requisitos: Node 18+

cd "../frontend"

npm install

Crear frontend/.env.local:

NEXT\_PUBLIC\_API\_URL=http://localhost:8000/api

Correr dev:

npm run dev

Abrir: http://localhost:3000

## **4) Flujo de ramas (no push a master)**

Ver ramas:

git branch -a

Cambiar a tu rama (ejemplos):

# si ya existe en remoto

git checkout DiegoSantis # o JonathanMorales / NicolasVergara

# sincronizar con master antes de trabajar

git pull --rebase origin master

Ciclo de trabajo:

git status

git add .

git commit -m "feat: mensaje claro del cambio"

git pull --rebase origin master

git push origin <tu-rama>

Abrir **Pull Request** desde tu rama → **master** (no hacer push directo a master).

## **5) Cosas que NO deben subirse (ya ignoradas)**

* backend/.venv/, backend/\_\_pycache\_\_/, backend/db.sqlite3
* frontend/node\_modules/, frontend/.next/, frontend/.turbo/
* backend/.env, frontend/.env.local

Si algo de eso se subió por error:

git rm -r --cached "FASE 2/Evidencias de Proyecto/backend/.venv"

git rm -r --cached "FASE 2/Evidencias de Proyecto/frontend/node\_modules"

git commit -m "chore: remove ignored folders"

git push

## **6) Pruebas rápidas**

* Backend levantado: http://127.0.0.1:8000/api/
* Login JWT (Postman):  
  + POST /api/auth/token/ → guarda access
* Frontend:  
  + /login → valida vacíos y hace POST a /api/auth/token/
  + /inventario (si está creado) → GET /materials/

## **7) Puertos ocupados (opcional)**

# Backend en otro puerto

python manage.py runserver 8001

# Frontend en otro puerto

npm run dev -- -p 3001

Eso es todo. Con estos comandos cualquier compañero clona, corre y trabaja en su rama sin romper master.