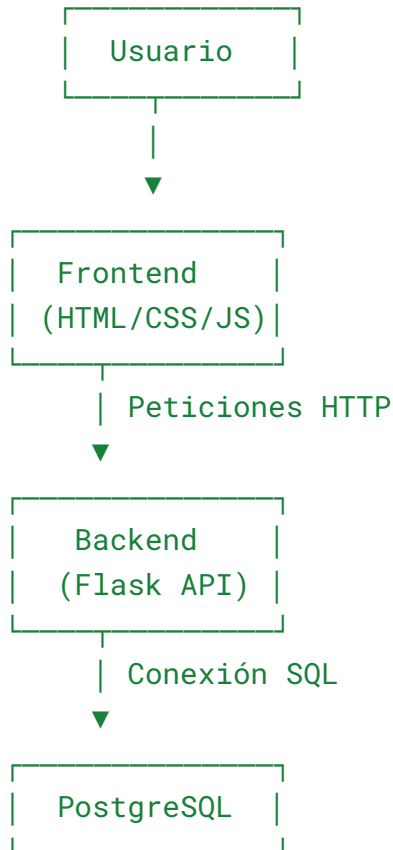


REPORTE DOCKER

Diagrama de Arquitectura de Servicios



- **Frontend:** Sirve archivos estáticos con HTML/CSS/JS.
- **Backend:** Una API Flask que procesa datos y se conecta con PostgreSQL.
- **Base de datos:** PostgreSQL inicializada con `init.sql`.

Configuración de Redes y Seguridad

Redes:

- Todos los servicios están definidos en un mismo archivo `docker-compose.yml`, lo que implica que están conectados a través de una **red bridge interna de Docker** (por defecto).

- El frontend se comunica con el backend, y el backend con la base de datos, usando nombres de servicio como hostname (por ejemplo, `db`, `backend`).

Seguridad:

- La base de datos no expone puertos al exterior (`5432` solo está disponible dentro de la red de Docker).
- Flask debería configurarse para no ejecutar en `debug` en producción.
- No se observan variables de entorno para contraseñas

Instrucciones de Despliegue

1. Requisitos previos:

- Tener Docker y Docker Compose instalados.

Estructura esperada del proyecto:

```
proyecto/
├─ docker-compose.yml
├─ init.sql
├─ backend/
│   ├─ app.py
│   ├─ db.py
│   ├─ Dockerfile
│   └─ requirements.txt
└─ frontend/
    ├─ index.html
    ├─ script.js
    ├─ style.css
    └─ Dockerfile
```

Despliegue:

En la raíz del proyecto (donde está `docker-compose.yml`), ejecutar:
`docker-compose up --build`

Acceso:

- El frontend probablemente estará en <http://localhost:8080>
- El backend estaría escuchando en otro puerto (usualmente [5000](#)).