

Projet DiabetesUFO

Plateforme web complète d'aide à la détection du diabète à partir de données utilisateurs. Architecture basée sur **FastAPI (Backend)** et **Streamlit (Frontend)**, conteneurisée avec **Docker Compose**.

Architecture

```
📁 Projet/
├─ Dockerfile.api # Image du backend FastAPI
├─ Dockerfile.front # Image du frontend Streamlit
├─ docker-compose.yml # Orchestration des deux services
├─ .env # Variables d'environnement
└─ README.md # Ce fichier
```

- **Backend (FastAPI)** : gère l'API, les modèles IA, et les endpoints.
 - **Frontend (Streamlit)** : interface utilisateur, appels à l'API.
-

Prérequis

Avant de lancer le projet, assure-toi d'avoir installé :

- [Docker](#)
- [Docker Compose](#)

Vérifie les versions :

```
docker --version
docker compose version
```

Configuration (.env)

Crée un fichier `.env` à la racine du projet avec le contenu suivant :

```
# Port utilisé par le backend FastAPI
PORT=8000
```

```
# URL interne de l'API utilisée par le frontend (ne pas modifier sauf si changement
de nom de service)
API_BASE_URL=http://api:8000
```

 Lancer le projet

Lancer le projet

1. Construire les images

```
docker compose build
```



2. Démarrer les conteneurs

```
docker compose up -d
```

Accès aux services :


- Frontend (Streamlit) : <http://localhost:8501>
- Backend (FastAPI) : <http://localhost:8000/docs>

Commandes utiles

 Action	 Commande
Démarrer le projet	<code>docker compose up</code>
Lancer en arrière-plan	<code>docker compose up -d</code>
Stopper le projet	<code>docker compose down</code>
Rebuild complet	<code>docker compose build --no-cache</code>
Voir les logs	<code>docker compose logs -f</code>

Vérification de santé (Healthchecks)

- FastAPI : `GET http://localhost:8000/healths`
- Streamlit : `GET http://localhost:8501/_stcore/health`

Les conteneurs ne se lancent que lorsque l'API est prête 

Nettoyage

Supprimer les conteneurs, réseaux et volumes :

```
docker compose down -v
```

Réseau Docker

Les deux services utilisent un réseau privé :

```
app-network (bridge)
```

Communication interne :

- le front appelle <http://api:8000>
- pas besoin d'exposer des ports internes

Documentation API

Une fois lancé : 🖱️ Accède à la documentation interactive Swagger : <http://localhost:8000/docs>

Astuce Développement

Pour un rechargement automatique lors des modifications locales :

- Active `--reload` dans le `Dockerfile.api` (FastAPI)
- Utilise le montage de volume :

```
volumes:  
- ./backend:/app
```

Licence

Projet pédagogique © 2025 — Utilisation libre à des fins d'apprentissage.

✨ Auteur Projet conçu pour démontrer une architecture multi-conteneurs avec FastAPI + Streamlit + Docker Compose. Par Olivier, Fidel et Anna :)