

## Manuel Technique – HealthPredict AI



**HealthPredict AI**

## 1. Introduction

Ce document décrit l'architecture technique, les dépendances et la procédure de déploiement de l'application **HealthPredict AI**.

Il est destiné aux **développeurs, intégrateurs et mainteneurs** du projet.

## 2. Architecture du projet

```
Healthpredict-AI-clean/
|
├─ app/                → Interface Streamlit (healthpredict_app.py)
├─ assets/             → Données et modèles
|   └─ data/           → Raw (OpenFDA), processed (labeled)
|   └─ models/         → Modèles IA entraînés (joblib)
|   └─ eval/           → Figures d'évaluation
├─ config/             → Fichiers de configuration (config.yaml, .streamlit/config.toml)
├─ data/               → Base SQLite (app.db), échantillons
├─ notebooks/          → Scripts d'évaluation (eval_healthpredict.py)
├─ scripts/            → Préparation données + entraînement IA
|   └─ build_processed_csv.py
|   └─ train_minimal_tfidf.py
|   └─ train_camembert_baseline.py
|   └─ download_assets.py
|   └─ ...
├─ requirements.txt     → Dépendances Python
├─ start.ps1           → Script de lancement (windows)
└─ hpdb.py             → Gestion SQLite
```

## 3. Dépendances principales

Langage : **Python 3.12+**

Librairies clés :

- **Streamlit** (UI)
- **Pandas / Numpy** (ETL & calculs)
- **Scikit-learn** (TF-IDF, modèles classiques)
- **Transformers / Torch** (CamemBERT)
- **SpaCy** (nettoyage linguistique)
- **Tesseract + pdf2image + PyPDF2** (OCR documents)
- **SQLite (hpdb.py)** (historique des prédictions)

Installation :

pip install -r requirements.txt

## 4. Configuration

### Fichiers clés :

- config/config.yaml : chemins des datasets et modèles.
- .streamlit/config.toml : paramètres UI (taille max messages, thème).
- .env : variables d'environnement (DB path, tokens Hugging Face).

### Variables utiles :

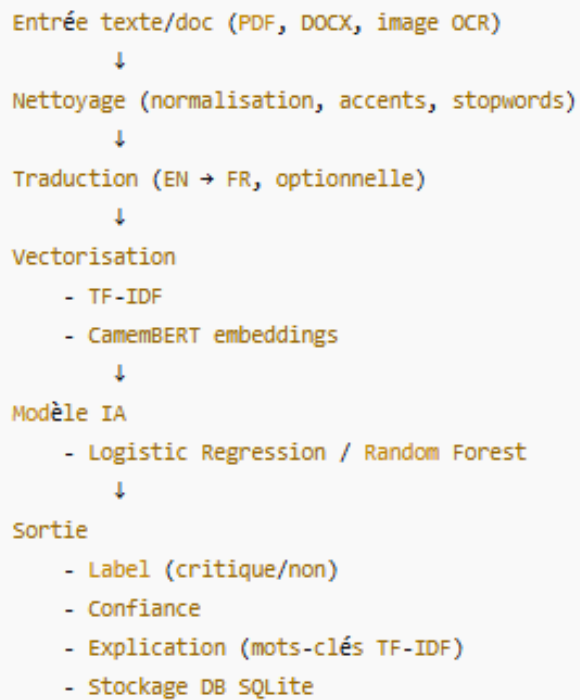
- HP\_AUTO\_DOWNLOAD=1 → téléchargement auto des modèles/données.
- HP\_DB=data/app.db → chemin de la base SQLite.
- HP\_USE\_CAMEMBERT=1 → active CamemBERT au lieu de TF-IDF.

## 5. Base de données

```
id          INTEGER PRIMARY KEY
ts          TIMESTAMP DEFAULT CURRENT_TIMESTAMP
source      TEXT    (texte, document, API...)
file_name   TEXT
input_text  TEXT
cleaned_text TEXT
model_type  TEXT (TF-IDF / CamemBERT)
label       TEXT (Critique / Non critique)
proba       FLOAT
detected_type TEXT
src_lang    TEXT (fr/en)
translated  BOOLEAN
top_keywords JSON
```

## 6. Pipeline IA

Schéma simplifié :



## 7. Entraînement des modèles

### TF-IDF (baseline)

python scripts/train\_minimal\_tfidf.py

Génère : assets/models/healthpredict\_model.joblib

### CamemBERT (avancé)

python scripts/train\_camembert\_baseline.py

Génère : assets/models/healthpredict\_camembert\_model.joblib

## 8. Déploiement

### Mode local (dev/test)

streamlit run app/healthpredict\_app.py

### Mode production (Docker possible)

- Utiliser requirements.txt pour installer dépendances.
- Stocker datasets & modèles sur Hugging Face (repo healthpredict-assets).

- Configurer pipeline CI/CD (GitHub Actions `.github/workflow/ci.yml`).

## 9. Maintenance & évolutions

- **Réentraînement** recommandé tous les 6 mois avec nouveaux incidents.
- **Améliorations possibles :**
  - Support multilingue (EN, FR).
  - API REST (FastAPI) pour intégration tierce.
  - Passage à une base PostgreSQL pour usage multi-utilisateurs.