

# Chien ou Chat ?



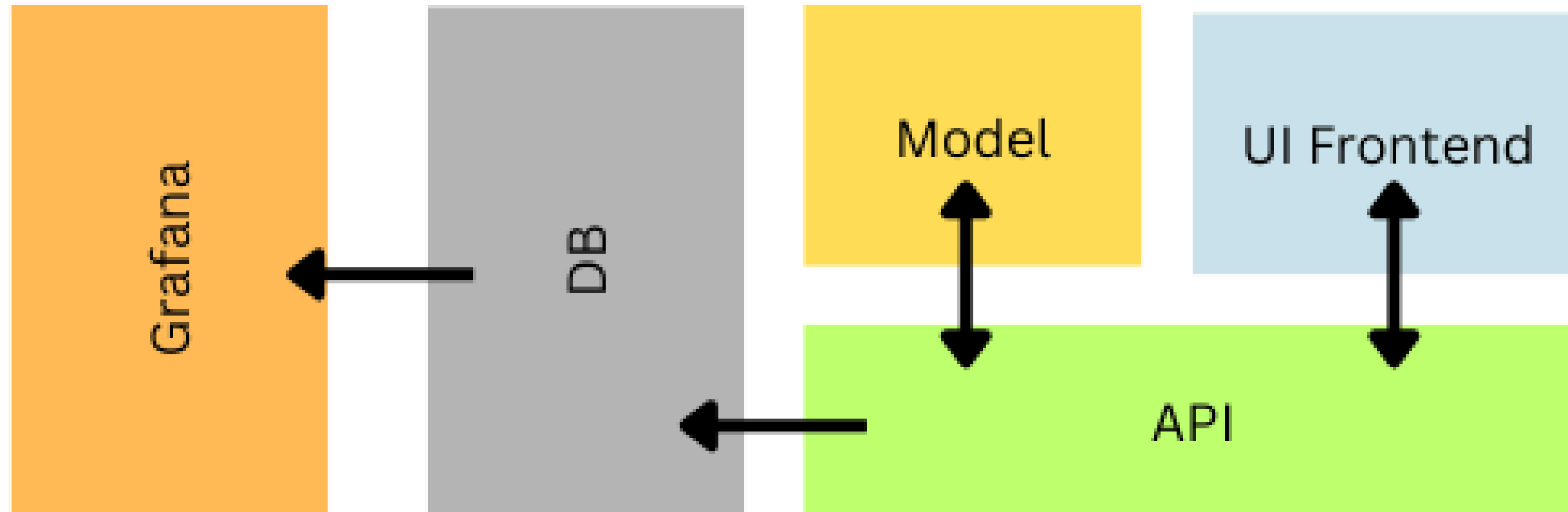
*Jonathan Caillaux*

# ***But du Projet***

*Améliorer une application existante*

- Stocker des information de monitoring en base
- Implementer une feedback loop
- Mettre en place un dashboard
- Établir une stratégie de ré-entraînement
- Améliorer la couverture des tests et CI
- Étendre la documentation existante

# *Architecture Globale*



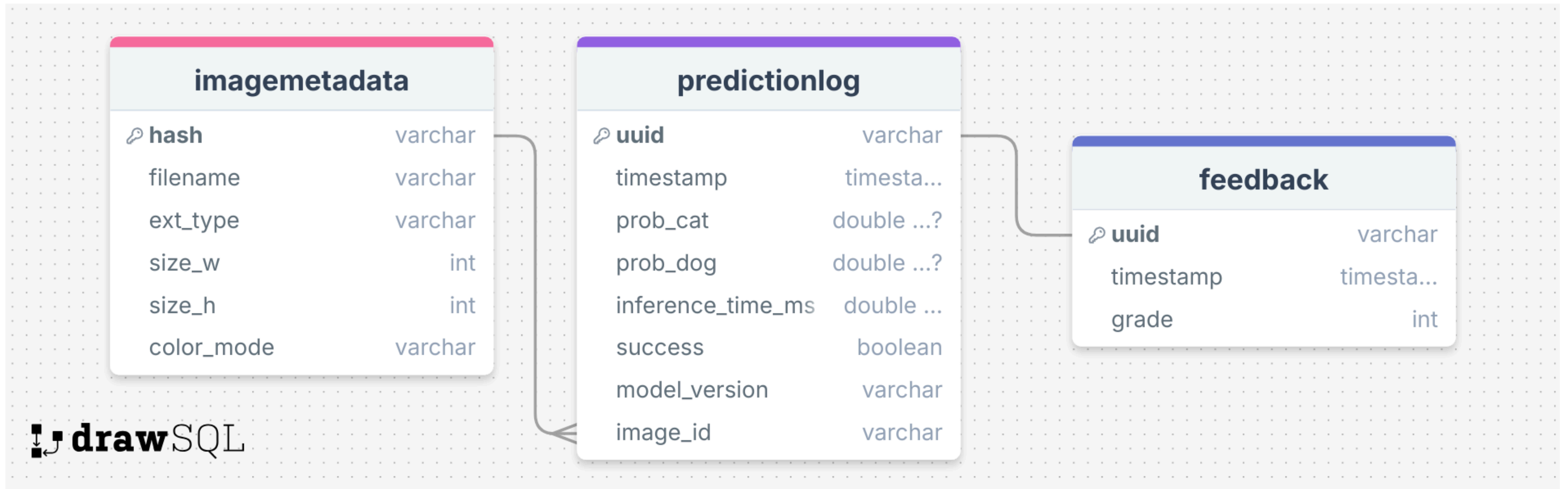
*DB : Postgres*

*API : FastAPI*

*UI : HTML (Jinja2 Templates)*

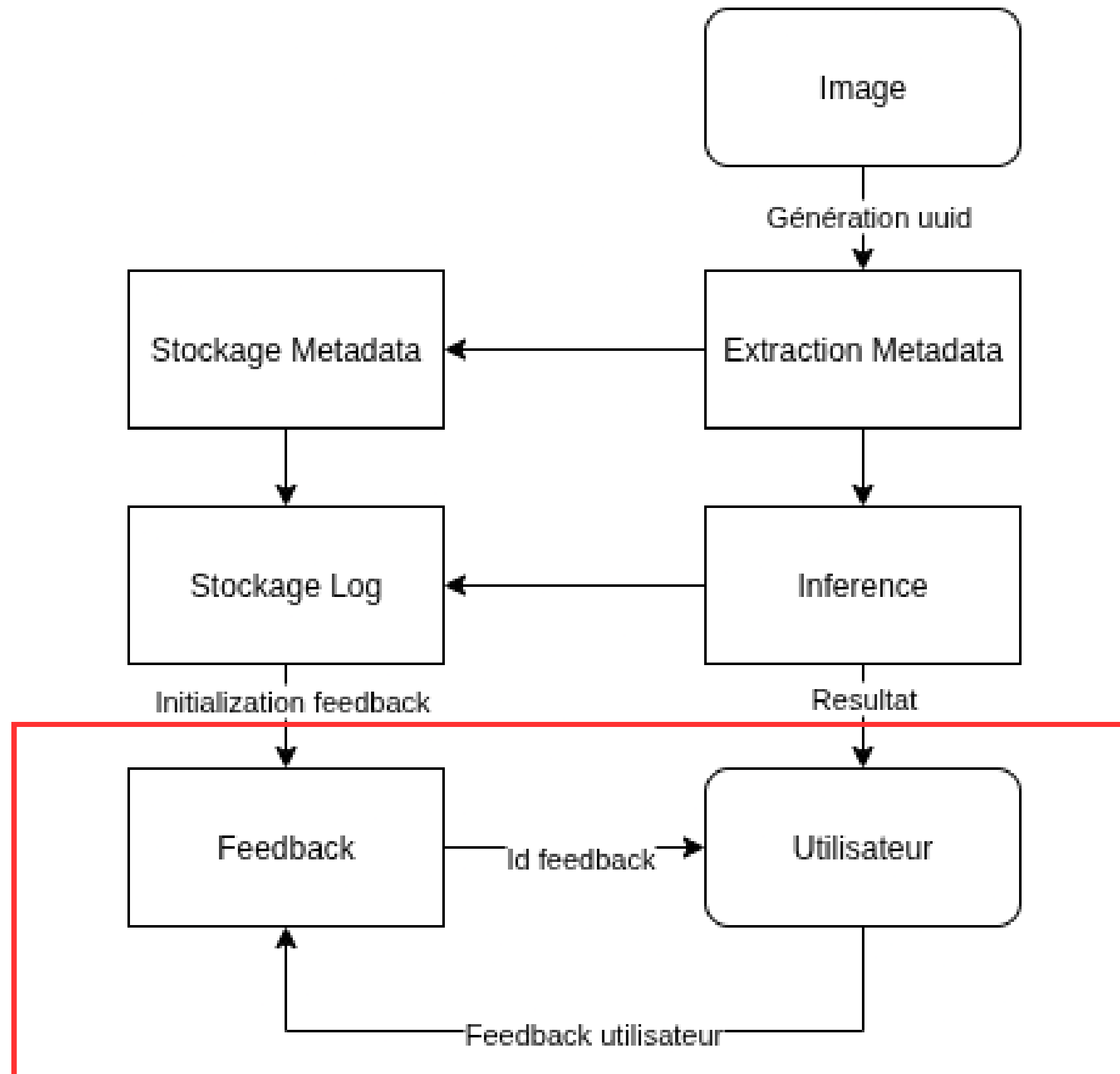
*Model : Keras(Tensorflow)*

# *Structure de la base SQL*



*3 tables (SoC)*

# ***Feedback-Loop***



# *Dashboard Grafana*



# ***Stratégie de ré-entraînement***

***Taux de satisfaction < 80% & taux de participation > 10 %***

*Alors*

*Un agent essaie de comprendre les causes*

*Ré-entraînement est Requis => augmentation du jeu d'entraînement avec des données correctement labélisées.  
Pas d'entraînement sur les données utilisateurs.*

# ***Couverture de test + CI***

- *Test des points d'API*
  - *Réponse des points d'API*
- *Test base de données (local)*
  - *Création + Suppression de tables*
  - *Lecture + écriture*

*Pas de test de performance ni de test sur le model*



# ***Documentation***

- *Actualisation du README*
- *Specification workflow git*
- *Notes RGPD*
- *Stratégie de ré-entraînement*

*cf dossier docs/*

# ***Conclusion & perspective***

## ***Conclusion***

*Fonctionnalités étendues :*

- Feedback loop
- Monitoring
- Dashboard

## ***Perspective***

- *Continuer l'implémentation de tests*
- *Création d'une base de test (CI)*
- *Améliorer l'architecture du code (front + back)*
- *Dockerisation*