

# **Cahier des Charges (CDC)**

## **PFE B3 Data & IA — 2025/2026**

### **Plateforme d'Annotation & Scoring de Contours Tumoraux (Human-in-the-Loop)**

Équipe	Minh Khang THAI, Léo MAHE, Paul WU, Mahdi BENNAMANE
Encadrant	M. Taalbi
Client / commanditaire	Skills4Mind (cadre pédagogique)
Version	v2.0
Date	27/01/2026

*Document interne — confidentiel (usage pédagogique)*

## Sommaire

- 1. Contexte et objectifs
- 2. Périmètre du projet
- 3. Parties prenantes et rôles
- 4. Parcours utilisateur et cas d'usage
- 5. Exigences fonctionnelles
- 6. Données et modèle SQL
- 7. Méthode de scoring (IA/métriques)
- 8. Architecture technique (Backend & DataViz)
- 9. Exigences non fonctionnelles (sécurité, green, souveraineté)
- 10. Plan projet, livrables et risques
- Annexes : Glossaire & API

## 1. Contexte et objectifs

Dans le cadre du PFE « IT for Green & Souveraineté Data », ce projet vise à concevoir une application web intégrant des briques full-stack (frontend, backend, base de données), du scoring basé sur des métriques, et une visualisation de données. L'application est pensée pour un usage pédagogique et d'entraînement.

### 1.1 Problématique

L'apprentissage de la détection/segmentation sur imagerie médicale nécessite de la pratique, ainsi qu'un retour objectif sur la qualité de l'annotation. Or, la correction manuelle par un expert est coûteuse et difficile à industrialiser.

### 1.2 Objectifs

Objectif principal (MVP) :

- Permettre l'annotation d'une région d'intérêt (ROI) sur une image médicale.
- Comparer l'annotation utilisateur à une référence (masque expert ou masque modèle) et calculer un score.
- Restituer un feedback visuel et chiffré, puis suivre la progression dans le temps.

Objectifs secondaires (optionnels) :

- Tableaux de bord avancés (comparaisons, classements, tendances).
- Assistant conversationnel pour expliquer les métriques et guider l'utilisateur (sans diagnostic).

### 1.3 Définition importante

Le système ne fournit pas de diagnostic médical. Il évalue uniquement la similarité entre une annotation et une référence (accord / métriques), dans un contexte d'entraînement.

## 2. Périmètre du projet

### 2.1 Périmètre MVP (in-scope)

- Chargement et visualisation d'images (DICOM ou images standards selon choix final).
- Outils d'annotation ROI (polygone et/ou ellipse) + édition.
- Conversion ROI → masque binaire + superposition (overlay).
- Calcul des métriques (Dice, IoU, précision, rappel).
- Stockage des résultats (annotation, métriques, temps, version de référence).
- Dashboard DataViz minimal (progression, distribution des scores).

### 2.2 Hors périmètre (out-of-scope)

- Aide à la décision clinique, diagnostic, ou recommandation médicale.
- Accès à des données patients non anonymisées.
- Certification dispositif médical / conformité clinique.
- Entraînement complet d'un modèle de segmentation à grande échelle (non requis pour le MVP).

## 2.3 Hypothèses et contraintes

- Les images et annotations de référence sont disponibles (dataset public ou jeux de données fournis/anonymisés).
- Les fichiers sont stockés et traités dans un environnement conforme à la souveraineté souhaitée (local/UE).
- La précision des métriques dépend de l'alignement des masques (même résolution/repère) et de la qualité de la référence.

## 3. Parties prenantes et rôles

Rôles principaux :

Rôle	Objectifs	Actions clés	Accès
Étudiant	S'entraîner et progresser	Annoter, lancer le scoring, consulter l'historique	Standard
Enseignant	Suivre et évaluer	Créer/assigner des cas, consulter stats par groupe	Élevé
Admin (option)	Gérer la plateforme	Gestion comptes, datasets, versions de référence	Complet

## 4. Parcours utilisateur et cas d'usage

### 4.1 Parcours utilisateur (MVP)

1. Connexion / création de compte (si activé).
2. Sélection d'un cas (image) dans une liste.
3. Annotation de la ROI dans le viewer.
4. Lancement du scoring et affichage des résultats (scores + overlay).
5. Sauvegarde automatique et consultation de l'historique / dashboard.

### 4.2 Cas d'usage (UC)

- UC1 — Annoter une image
- UC2 — Comparer à la référence et calculer les métriques
- UC3 — Consulter les résultats d'un cas
- UC4 — Suivre sa progression (courbes / distributions)
- UC5 — (Option) Créer une session/examen blanc et partager un code

## 5. Exigences fonctionnelles

Notation des priorités : Must (indispensable), Should (important), Could (bonus).

Fonction	Description	Priorité	Critère d'acceptation
Viewer image	Afficher image (zoom/pan, slices si DICOM)	Must	Navigation fluide, image lisible
Annotation ROI	Dessiner/éditer polygone ou ellipse	Must	ROI sauvegardable, éditable
ROI→masque	Rasterisation ROI en masque binaire	Must	Masque aligné à l'image
Scoring	Calcul Dice, IoU, précision, rappel	Must	Scores cohérents sur cas test
Overlay erreurs	Visualiser FP/FN sur l'image	Should	Zones mises en évidence
Historique	Lister tentatives et scores par utilisateur	Must	Historique filtrable par date/cas
Dashboard	Courbes et distributions de scores	Should	Graphiques lisibles, requêtes rapides
Exam blanc	Code de session + sélection de cas	Could	Étudiants rejoignent via code

## 6. Données et modèle SQL

### 6.1 Stockage : données structurées vs fichiers

La base SQL stocke les métadonnées (utilisateurs, cas, annotations, scores). Les fichiers lourds (images, masques, overlays) sont stockés dans un stockage objet (local/MinIO/S3), référencés par des URL/chemins en base.

### 6.2 Schéma de données (MVP)

Table	Champs clés	Description
users	id, email, role, created_at	Comptes et rôles
cases	id, title, modality, created_at	Cas d'entraînement
images	id, case_id, storage_url, format,	Fichiers image et métadonnées

	metadata_json	
references	id, case_id, type, storage_url, version	Masque de référence (expert/modèle)
annotations	id, user_id, case_id, roi_json, created_at, duration_sec	ROI utilisateur (vectoriel)
scores	id, annotation_id, dice, iou, precision, recall, fp_area, fn_area	Métriques calculées

### 6.3 Indexation (performance)

- Index sur annotations(user\_id, created\_at) pour la progression.
- Index sur scores(annotation\_id) et images(case\_id).
- Pagination + filtres côté SQL pour limiter les transferts (sobriété).

## 7. Méthode de scoring (IA/métriques)

### 7.1 Pipeline

6. 1) L'utilisateur dessine une ROI (polygone/ellipse).
7. 2) La ROI est convertie en masque binaire (même taille que l'image).
8. 3) Le masque utilisateur est comparé au masque de référence.
9. 4) Les métriques sont calculées et stockées ; un overlay FP/FN est généré.

### 7.2 Métriques (définitions courtes)

- IoU (Intersection over Union) : surface d'intersection / surface d'union.
- Dice :  $2 \times \text{intersection} / (\text{surface masque utilisateur} + \text{surface référence})$ .
- Précision :  $\text{TP} / (\text{TP} + \text{FP})$ .
- Rappel :  $\text{TP} / (\text{TP} + \text{FN})$ .

### 7.3 Limites

- La qualité du score dépend de la référence (ground truth).
- Un mauvais alignement (résolution/repère) fausse les métriques.
- Les annotations proches du bord peuvent être sensibles au choix ROI (ellipse vs polygone).

## 8. Architecture technique (Backend & DataViz)

### 8.1 Backend (outils principaux)

- FastAPI : API REST, validation des entrées, documentation OpenAPI.
- SQLAlchemy + Alembic : accès PostgreSQL et migrations.
- NumPy + OpenCV : génération des masques, calcul des métriques, overlays.
- Stockage objet (MinIO/S3) : images, masques, artefacts de scoring.

## 8.2 DataViz (dashboard)

- ECharts (ou Recharts) : courbes de progression (Dice/IoU), histogrammes de distribution, comparaisons.
- Endpoints API /stats : agrégations SQL (moyenne, médiane, percentiles) pour alimenter les graphiques.

## 8.3 Diagramme (à insérer)

À insérer dans le rendu final : schéma blocs Frontend ↔ API ↔ (PostgreSQL + Storage).

# 9. Exigences non fonctionnelles

## 9.1 Sécurité et conformité

- Authentification JWT et rôles (Étudiant / Enseignant / Admin).
- Logs techniques sans données sensibles ; suppression/rotation des logs.
- Données : priorité à des datasets publics ou anonymisés (pas de données patients identifiantes).

## 9.2 Performance & qualité

- Temps de scoring cible : inférieur à 2 secondes par cas sur machine de démo (à ajuster selon format).
- Tests unitaires sur les fonctions de métriques (cas simples contrôlés).
- Versioning : tracer la version de référence (expert/modèle) utilisée pour chaque score.

## 9.3 Green IT & souveraineté data

- Sobriété : limiter la taille des transferts (compression, cache, pagination).
- Souveraineté : privilégier stockage local/UE ; dépendances open-source ; possibilité de déployer on-prem.
- Traçabilité des sources : documenter l'origine des images/annotations et les droits d'usage.

# 10. Plan projet, livrables et risques

## 10.1 Roadmap (sprints indicatifs)

Sprint	Objectifs	Livrables
S1	Setup repo, Docker, DB, auth	Repo, Compose, schéma DB
S2	Viewer + import cas	Écran viewer + liste de cas
S3	Annotation ROI + sauvegarde	Outils ROI + table annotations
S4	ROI→masque + scoring + résultats	Métriques + overlay + page résultats
S5	Dashboard + optimisations SQL	Graphiques + endpoints /stats
S6	Polish, tests, documentation, démo	README, démo, slides

## 10.2 Livrables attendus

- CDC (ce document) + maquettes Figma.
- Repository Git (frontend, backend, infra) + documentation d'installation.
- Jeu de démo (quelques cas) + démonstration scoring + dashboard.

## 10.3 Risques & parades

Risque	Impact	Parade
Référence non fiable / incomplète	Scores incohérents	Utiliser dataset public validé, tests sur cas simples
Complexité DICOM	Retard viewer	MVP sur images standards puis extension DICOM
Temps de scoring trop long	UX dégradée	Optimiser NumPy, pré-calculs, limiter résolution pour démo
Problèmes de stockage	Perte de fichiers	MinIO + sauvegarde + URLs versionnées

## Annexes

### A. Glossaire

- ROI : Region of Interest (zone annotée).
- Masque binaire : image 0/1 représentant une zone.
- TP/FP/FN : True Positive / False Positive / False Negative.
- IoU/Dice : métriques de similarité entre masques.

### B. Endpoints API (MVP)

Endpoint	Méthode	Description
/cases	GET	Lister les cas disponibles
/cases/{id}	GET	Détails d'un cas + liens fichiers
/annotations	POST	Soumettre une ROI (roi_json)
/verify	POST	Calculer métriques vs référence
/scores/{annotation_id}	GET	Récupérer les métriques d'une tentative
/stats	GET	Agrégations pour dashboard (par user/date/case)