

# TEST TECHNIQUE – DÉVELOPPEUR IA

## Objectif du test

- Comprendre un besoin métier concret
- Concevoir une solution IA réaliste et applicable
- Structurer une architecture claire
- Proposer une implémentation simple mais cohérente

## Contexte

Soit une plateforme de mise en relation entre :

- Des recruteurs proposant des missions
- Des candidats disposant de CV et de compétences

L'objectif est d'améliorer la qualité du Matching et la prise de décision grâce à l'IA

### **A) Matching intelligent (obligatoire)**

#### Problématique

Comment calculer un score de compatibilité entre :

- Un CV candidat
- Une offre de mission

#### Attendus

Fournir :

- Une liste des critères utilisés (exemple : compétences, expérience, localisation, disponibilité)
- Une méthode de calcul du score (logique ou formule)
- Un exemple chiffré de score
- Une explication simple du score pour un recruteur

#### Livrable

- Pseudo-code ou code simplifié du calcul de score
- Explication écrite claire

### **B) Analyse automatique des CV (obligatoire)**

Extraire automatiquement les informations clés d'un CV

#### Attendus

Decrire :

- Le pipeline de traitement du CV (upload → OCR → extraction → structuration)
- Les informations extraites (nom, compétences, expériences, diplômes)
- La gestion des CV mal formatés ou incomplets

### **Livrable**

- Schéma simple du pipeline
- Description technique étape par étape

### **C) Recommandation intelligente (obligatoire)**

#### **Cas d'usage**

- Recommander des candidats à un recruteur
- Recommander des missions à un candidat

#### **Attendus**

Expliquer :

- La logique de recommandation utilisée
- La différence entre matching et recommandation
- Comment le système peut s'améliorer avec le temps

### **Livrable**

- Description fonctionnelle
- Pseudo-code ou logique algorithmique

### **D) Architecture technique IA (obligatoire)**

#### **Attendus**

Proposer une architecture incluant :

- Un service IA dédié
- Une base de données (classique et/ou vectorielle)
- Une API exposée au backend
- Une séparation claire des responsabilités

#### **Contraintes**

- Architecture simple et réaliste
- Scalable
- Explicable

### **Livrable**

- Schéma d'architecture (même simple)
- Description des flux de données

### **E) Choix technologiques (obligatoire)**

Préciser :

- Langage principal utilisé
- Librairies IA ou NLP
- Type de modèle utilisé (règles, ML, LLM, hybride)
- Justification des choix

**Livrables finaux**

- Document unique (PDF ou Markdown)
- Schémas intégrés ou séparés
- Dépôt Git (optionnel)