

Plan de Tests pour JOB MATCHER

Étape 1 : Types et Techniques de Tests pour Chaque Module

1. Module ACCÈS :

- **Tests fonctionnels :**
 - Vérification des fonctionnalités de navigation et d'authentification.
 - Test des exigences UX avec des scénarios utilisateurs variés.
- **Tests de performance :**
 - Test de charge pour vérifier la capacité du portail à gérer un grand nombre d'utilisateurs simultanés.
- **Tests de sécurité :**
 - Vérification des failles XSS sur les commentaires authentifiés.
 - Contrôle d'accès pour éviter tout accès non autorisé.
- **Tests de compatibilité :**
 - Validation du fonctionnement sur plusieurs navigateurs et appareils (desktop, mobile, tablette).
- **Techniques recommandées :**
 - Test exploratoire pour couvrir les cas inattendus.
 - Automatisation pour les tests de régression fréquents.

2. Module CV+ :

- **Tests de confidentialité et de sécurité :**
 - Tests contre les injections SQL.
 - Vérification du chiffrement des données personnelles.
 - Fuzzing pour détecter des vulnérabilités.
- **Tests fonctionnels :**
 - Vérification du téléchargement et de la gestion des CV et lettres de motivation.
- **Tests d'intégration :**
 - Validation des interactions entre l'interface utilisateur et le backend.
- **Techniques recommandées :**
 - Automatisation des tests critiques pour garantir une couverture continue.

3. Module MATCHER-OFFRES :

- **Tests fonctionnels :**
 - Validation des règles de gestion élaborées pour le filtrage des offres.
 - Tests des processus de candidature en ligne.
- **Tests de performance :**
 - Validation de la vitesse de filtrage pour un grand volume de données.
- **Tests de bout en bout (E2E) :**

- Simulation du parcours d'un utilisateur déposant un CV et consultant des offres.
 - **Techniques recommandées :**
 - TDD pour gérer l'ajout ou la modification des règles de gestion.
 - 4. **Module MATCHER-TALENTS :**
 - **Tests fonctionnels :**
 - Vérification des notifications et de la visualisation des CV adaptés.
 - **Tests d'intégration :**
 - Validation de l'interconnexion avec le backend et le système de gestion des notifications.
 - **Tests utilisateurs (BDD) :**
 - Recueil de retours auprès des recruteurs pour s'assurer que les besoins sont couverts.
 - **Techniques recommandées :**
 - Automatisation des scénarios communs.
 - Tests exploratoires pour des retours qualitatifs.
 - 5. **Module SUPERVIZ (Backend uniquement) :**
 - **Tests d'intégration avec l'IA :**
 - Validation de la connexion avec le composant IA.
 - **Tests fonctionnels :**
 - Vérification de la génération de rapports en langage naturel.
 - **Tests de performance :**
 - Mesure de la rapidité et de l'efficacité de l'analyse des CV.
 - **Techniques recommandées :**
 - Utilisation de fuzzing pour tester des cas limites.
-

Étape 2 : Bonnes Pratiques à Mettre en Place

1. **Pratiques générales :**
 - Implémentation d'un pipeline CI/CD (GitLab) pour exécuter les tests de manière automatique.
 - Mise en place de code reviews systématiques avant toute intégration.
 - Adoption du développement piloté par les tests (TDD) pour minimiser les bugs en amont.
2. **Tests automatisés :**
 - Automatisation des tests unitaires, d'intégration et de régression pour réduire les efforts manuels.
 - Mise en place de métriques comme la couverture de code pour évaluer l'efficacité des tests.

3. Sécurité et qualité :

- Analyse statique du code pour détecter des erreurs avant l'exécution.
- Formation des développeurs aux principes de sécurité (OWASP).

4. Collaboration et retours :

- Organisation de tests utilisateurs pour les modules à forte interaction (ACCÈS, MATCHER-TALENTS).
- Boucles de feedback régulières entre équipes de test et de développement.

5. Performance et robustesse :

- Tests de charge réguliers sur les modules critiques (ACCÈS, SUPERVIZ).
- Simulation de scénarios réalistes pour évaluer la résilience du système.