**Plan de test – Fonction getUtilisateurs**

**1 - Identification du test**

**Nom** : Test de récupération de la liste des utilisateurs  
**Numéro** : T10

**2 - Référence du module testé**

routes/pgs.py – Fonction getUtilisateurs

**3 - Objectif du test**

Vérifier que la route retourne :

* Tous les utilisateurs de la base
* Avec tous leurs champs visibles, y compris le mot de passe en clair via le schéma RecupUtilisateur
* Et qu’une erreur est levée si aucun utilisateur n’est trouvé

**4 - Procédure du test**

* **Initialisation** : Remplir ou vider la table Utilisateur
* **Lancement** : Requête GET sur /pgs/utilisateur/
* **Observation** : Vérifier le code de retour et les données retournées :
  + Liste d’utilisateurs avec : id, nom, prenom, identifiant, role, date\_de\_naissance, mot\_de\_passe, id\_classe

**5 - Résultats attendus**

| **N° Test** | **Condition** | **BDD Préparée** | **Résultat attendu** | **Statut attendu** |
| --- | --- | --- | --- | --- |
| 1 | Aucun utilisateur | Table Utilisateur vide | 404 Aucun utilisateur | Erreur |
| 2 | Utilisateurs présents | Table remplie avec plusieurs utilisateurs | 200 Liste complète des utilisateurs (avec mot de passe déchiffré) | Succès |

**6 - Moyens à mettre en œuvre**

* **Logiciels** : FastAPI, SQLAlchemy, Pytest
* **Matériel** : Poste de développement, base test
* **Préconditions** :
  + Table Utilisateur bien définie avec tous les champs
  + Schéma RecupUtilisateur prêt pour le from\_orm
  + Mot de passe visible ou déchiffré directement selon l’implémentation de RecupUtilisateur