**Plan de test – Fonction activiteSalle**

**1 - Identification du test**

Nom : Test de récupération de l’activité d’une salle en temps réel (corrigé)  
Numéro : T15

**2 - Référence du module testé**

routes/psw.py – Fonction activiteSalle

**3 - Objectif du test**

Vérifier que cette route renvoie correctement :

* Les réservations de la salle
* Les utilisateurs ayant badgé récemment (s’il y en a)
* Pas d'erreur si aucun utilisateur n'a badgé (renvoi liste vide)

Et toujours lever une erreur 404 si :

* Salle inexistante
* Aucune réservation
* Réservations sans utilisateur
* Utilisateurs manquants

**4 - Procédure du test**

* **Initialisation :** Préparer en BDD une salle, des réservations, utilisateurs, badges et logs récents ou non
* **Lancement :** Requête GET sur /psw/salle/{id\_salle}/activite
* **Observation :** Vérifier le JSON retourné ou les erreurs

**5 - Résultats attendus**

| **N° Test** | **Condition** | **BDD Préparée** | **Résultat attendu** | **Statut attendu** |
| --- | --- | --- | --- | --- |
| 1 | Salle inexistante | Pas de salle avec cet ID | 404 Salle non trouvée | Erreur |
| 2 | Aucune réservation | Salle existe, mais pas de EDTSalle | 404 Aucune réservation trouvée | Erreur |
| 3 | Réservation sans utilisateur | Réservation avec id\_utilisateur = None | 404 Réservation sans utilisateur associée | Erreur |
| 4 | Utilisateur réservation introuvable | Utilisateur non trouvé en BDD | 404 Utilisateur (ID X) introuvable | Erreur |
| 5 | Aucun badge récent | Pas de badge scanné dans l’heure | 200 Réservations OK + utilisateurs\_derniere\_heure = [] + nombre\_utilisateurs = 0 | Succès |
| 6 | Badge introuvable | UID de log inexistant | 404 Badge UID X introuvable | Erreur |
| 7 | Badge non associé | Badge trouvé mais sans utilisateur lié | 404 Badge UID X non associé | Erreur |
| 8 | Utilisateur badge introuvable | Badge existant mais utilisateur lié inexistant | 404 Utilisateur ID X introuvable pour badge UID | Erreur |
| 9 | Succès complet | Réservations + logs récents + utilisateurs valides | 200 JSON avec réservations + utilisateurs + nombre\_utilisateurs > 0 | Succès |

**6 - Moyens à mettre en œuvre**

* **Logiciels :** FastAPI, SQLAlchemy, Pytest
* **Matériel :** Poste de développement
* **Préconditions :**
  + Tables Salle, EDTSalle, Utilisateur, Badge, Log, Equipement bien configurées
  + Route /psw/salle/{id\_salle}/activite opérationnelle
  + Données insérées avec cohérence