

RAPPORT D'ANOMALIE

Gestion des Défauts (Defect Lifecycle)

Équipe QA :

Mohamed Dhia Eddine Thabet *Test Manager*
Mohamed Aziz Dridi *Tester*

Novembre 2025

1 Anomalie 1 : Instabilité des Données

Champ	Détails
ID Défaut	BUG-BACK-001
Titre	Duplication d'IDs et perte de données sous charge.
Sévérité	CRITIQUE
Priorité	HAUTE
Composant	TodoRepository.java
DéTECTé via	Test de Performance JMeter

Description Technique

La variable `idCounter` est incrémentée de manière non atomique (`++`). Lors du test de charge (50 utilisateurs simultanés), plusieurs tâches ont reçu le même ID, provoquant l'écrasement des données précédentes en mémoire. Cela constitue une violation critique de l'intégrité des données (Data Integrity Violation).

Preuve / Log

```

2025-11-29 23:45:12.102
INFO [nio-8080-exec-3]
c.d.t.s.TodoService      :
Processing request: Create Task "LoadTest_User_03"

2025-11-29 23:45:12.102
INFO [nio-8080-exec-7]
c.d.t.s.TodoService      :
Processing request: Create Task "LoadTest_User_07"

2025-11-29 23:45:12.105
DEBUG [nio-8080-exec-3]
c.d.t.r.TodoRepository   :
Assigning ID. Current counter value: 142

2025-11-29 23:45:12.105
DEBUG [nio-8080-exec-7]
c.d.t.r.TodoRepository   :
Assigning ID. Current counter value: 142 <-- CONFLIT DÉTECTÉ

2025-11-29 23:45:12.106
INFO [nio-8080-exec-3]
c.d.t.r.TodoRepository   :
Saved TodoItem {id=142, task='LoadTest_User_03'}

```

FIGURE 1 – Évidence du défaut de concurrence

2 Anomalie 2 : Le "Zombie Item"

Champ	Détails
ID Défaut	BUG-LOGIC-002
Titre	Réapparition inattendue d'une tâche supprimée.
Sévérité	MAJEURE
Priorité	MOYENNE
Composant	TodoRepository.java (Méthode <code>fetchAllTodos</code>)
DéTECTÉ via	Test E2E Cypress (TC-05)

Description Fonctionnelle

Le backend contient une règle métier explicite qui recrée une tâche par défaut "Click to edit task name" si la liste devient vide.

- **Conséquence :** L'utilisateur ne peut jamais avoir une "Inbox Zero". S'il supprime tout, une tâche revient. Cela contredit les principes d'ergonomie standards (UX) et peut causer de la confusion.

Preuve Visuelle

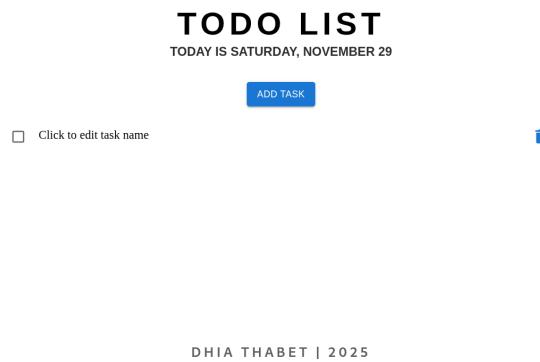


FIGURE 2 – La tâche réapparaît après suppression (Cypress)

3 Anomalie 3 : Risque de Boucle Infinie (Frontend)

Champ	Détails
ID Défaut	BUG-FRONT-003
Titre	Gestion risquée du Hook <code>useEffect</code> (Infinite Render).
Sévérité	MOYENNE
Priorité	BASSE
Composant	App.js
Détecté via	Analyse Statique SonarQube

Description Technique

Dans le fichier `App.js`, le hook `useEffect` qui charge les données possède `todoItems` dans son tableau de dépendances.

- **Problème :** Si `setTodoItems` est appelé à l'intérieur de l'effet (ce qui est le cas via `fetch`), cela modifie `todoItems`, ce qui redéclenche l'effet, créant potentiellement une boucle infinie de requêtes réseau.
- **Impact :** Risque de saturation du navigateur client et surcharge inutile du backend.

Recommandation de Correction

Retirer `todoItems` du tableau de dépendances ou utiliser une fonction de mise à jour fonctionnelle pour briser le cycle.