

Pour assurer la robustesse de votre application, il est crucial de tester des scénarios qui vont au-delà du fonctionnement nominal. Voici quelques idées :

1. Tests de Cas Limites et Erreurs de Fichiers/Répertoires

- **Chemins Invalides :**
 - Tentez de synchroniser vers une source ou une destination qui n'existe pas.
 - Utilisez des chemins avec des caractères spéciaux (!@#\$%^&*()), espaces, accents).
 - Utilisez des chemins très longs.
 - Utilisez des chemins réseau (si l'application est censée les supporter).
- **Permissions :**
 - Tentez de synchroniser vers une destination où l'utilisateur n'a pas les droits d'écriture.
 - Tentez de synchroniser depuis une source où l'utilisateur n'a pas les droits de lecture.
- **Fichiers Verrouillés/En Cours d'Utilisation :**
 - Modifiez un fichier dans le répertoire source pendant qu'une synchronisation est en cours.
 - Laissez un fichier ouvert par une autre application (ex: un document Word, un fichier vidéo) dans le répertoire source et lancez la synchro.
- **Fichiers Spéciaux :**
 - Synchronisez des fichiers de taille zéro.
 - Synchronisez des fichiers très volumineux (plusieurs Go).
 - Synchronisez un très grand nombre de petits fichiers.
 - Synchronisez des répertoires avec une très grande profondeur d'arborescence.
- **Disque Plein :**
 - Simulez un disque de destination plein ou presque plein pendant une synchronisation.

2. Tests de Concurrency et d'Interruption

- **Arrêt Brutal :**
 - Arrêtez le frontend ou le backend en plein milieu d'une synchronisation (via le gestionnaire de tâches ou Ctrl+C). Que se passe-t-il au redémarrage ? L'état est-il récupéré correctement ?
 - Arrêtez le système d'exploitation pendant une synchro.
- **Déconnexion Réseau :**
 - Si la destination est un partage réseau, coupez la connexion réseau pendant la synchronisation.
- **Plusieurs Synchronisations Concurrentes :**
 - Lancez deux synchronisations différentes (sources/destinations différentes) en même temps. Assurez-vous qu'elles ne s'interfèrent pas et que leurs états sont correctement suivis.
 - Lancez deux instances du frontend et tentez de contrôler la même

synchronisation ou des synchronisations différentes.

- **Modification de la Source/Destination pendant la Synchro :**
 - Renommez ou supprimez un fichier/répertoire dans la source ou la destination pendant qu'une synchronisation est en cours.

3. Tests de Performance

- **Volumes de Données :**
 - Testez avec des répertoires contenant des dizaines de milliers de fichiers.
 - Testez avec des répertoires contenant des fichiers de très grande taille.
- **Fréquence Élevée :**
 - Définissez une fréquence de synchronisation très courte (ex: 1 minute) pour voir si le système gère bien les cycles rapides.

4. Tests d'Interface Utilisateur (Frontend)

- **Saisie Invalide :**
 - Entrez du texte non numérique dans les champs numériques (fréquence, versions en cache).
 - Laissez des champs obligatoires vides.
- **Clics Répétés :**
 - Cliquez rapidement plusieurs fois sur "Start" ou "Stop".
- **Redimensionnement de la Fenêtre :**
 - Assurez-vous que l'interface reste utilisable et esthétique lors du redimensionnement de la fenêtre.
- **Chargement/Sauvegarde de Configuration :**
 - Tentez de charger un fichier qui n'est pas une configuration JSON valide.
 - Tentez de sauvegarder une configuration avec un nom de fichier invalide.

En couvrant ces scénarios, vous pourrez identifier et corriger de nombreux points faibles, rendant votre application beaucoup plus fiable.