# Exercice : Opérations CRUD avancées et sécurisées avec relations et validation

## Objectif

Améliorer le système CRUD existant pour inclure la validation, la gestion des erreurs et les relations entre les modèles.

## Instructions

* Définir un modèle générique :
* Mettre à jour l'interface `Model` pour inclure des timestamps `createdAt` et `updatedAt`.
* Implémenter un dépôt générique :
* Étendre la classe `Repository` pour inclure une logique de validation pour la création et la mise à jour des éléments.
* Ajouter la gestion des erreurs aux méthodes pour fournir des messages d'erreur significatifs.
* Définir des modèles spécifiques avec relations :
* Créer un modèle `Post` qui étend `Model` et inclut `title`, `content` et `authorId` (référençant un `User`).
* Modifier le modèle `User` pour inclure une liste d'IDs de `Post`.
* Créer des dépôts pour les modèles spécifiques :
* Instancier des dépôts pour `User` et `Post`.
* Ajouter des méthodes pour récupérer les posts d'un utilisateur spécifique.
* Implémenter la sécurité des types et les types utilitaires :
* S'assurer que les méthodes du dépôt n'acceptent et ne retournent que les types appropriés.
* Utiliser des types utilitaires pour rendre les propriétés en lecture seule là où c'est applicable.
* Ajouter la validation et la gestion des erreurs :
* Implémenter des vérifications de validation dans les méthodes du dépôt.
* Ajouter une gestion des erreurs appropriée pour assurer la robustesse des opérations CRUD.