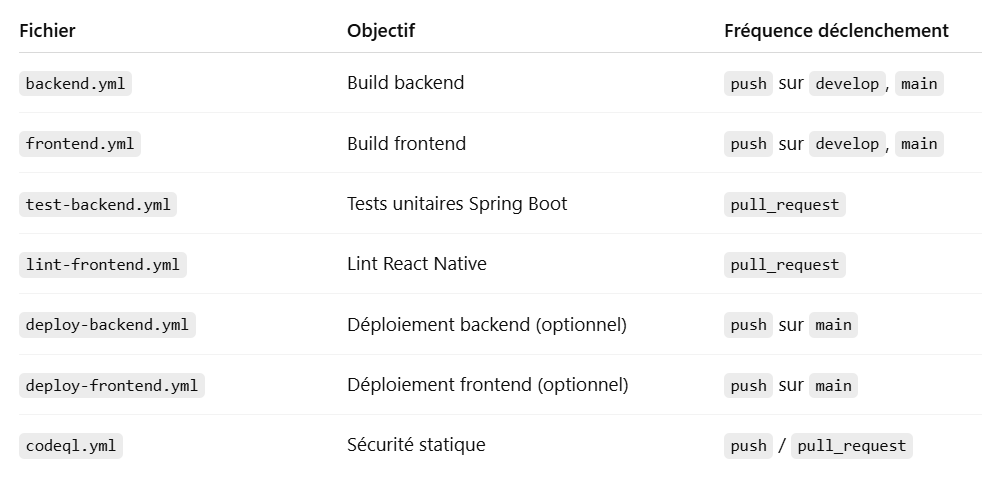
Pipeline CI/CD :

Les fichiers backend.yml et frontend.yml permettent de vérifier que le backend Spring Boot et le frontend React Native se construisent correctement après chaque push sur les branches develop et main

Workflows :



test-backend.yml – Tests automatisés pour le backend

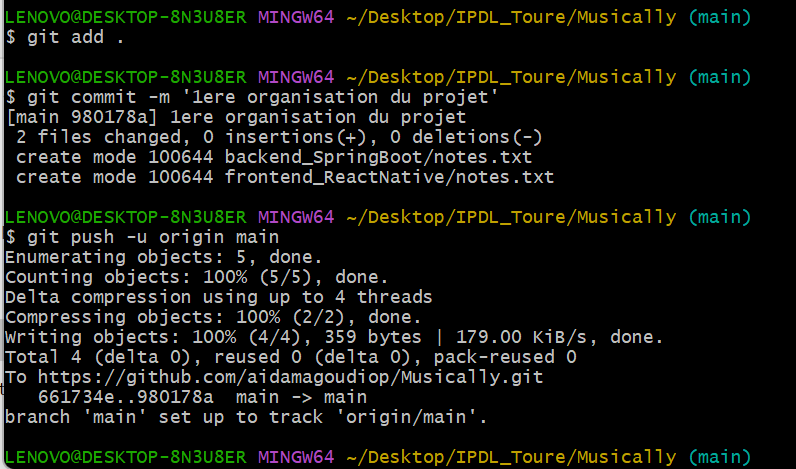
Objectif : Lancer les tests unitaires (JUnit) sur le backend.

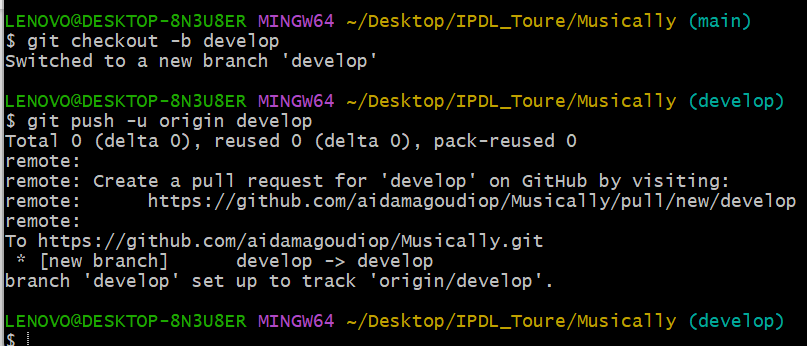
lint-frontend.yml – Linting du code React Native

Objectif : Vérifier la qualité syntaxique du code frontend.

Le front React Native **doit avoir un script lint dans**

**package.json !!!!!!**





**Branches :**

- main : branche principale et stable. Elle contient le code prêt pour la production.

- develop : branche de développement. Elle sert à intégrer les fonctionnalités avant qu’elles ne soient fusionnées dans main.

Tu travailles au quotidien sur develop, puis tu fais un merge vers main quand une version stable est prête.

**Commandes Workflow Git :**

**- pour chaque développeur**

# 1. Cloner le dépôt

git clone https://github.com/aidamagoudiop/Musically.git

# 2. Aller dans le dossier du projet

cd Musically

# 3. Basculer sur la branche develop

git checkout develop

**- Frontend :**

# Créer une branche de fonctionnalité (ex: ajout de la page login)

git checkout -b feat/frontend-login

# Travailler uniquement dans le dossier frontend/

cd frontend/

# Ajouter les modifications

git add frontend/

# Committer

git commit -m "Ajout de la page de login"

# Pousser la branche

git push -u origin feat/frontend-login

**- backend :**

# Créer une branche de fonctionnalité (ex: API d'authentification)

git checkout -b feat/backend-auth

# Travailler uniquement dans le dossier backend/

cd backend/

# (ajout de contrôleurs, services, etc.)

# Ajouter les modifications

git add backend/

# Committer

git commit -m "Ajout de l’API d’authentification"

# Pousser la branche

git push -u origin feat/backend-auth

**Apres modification de codes**

# 2. S’assurer d'être à jour sur develop:

git checkout develop

git pull origin develop , avant de faire git checkout -b …..

Forcer GitHub Actions à ne build que le dossier modifié grâce à paths dans tes .yml, pour eviter de lancer un build backend quand un dev frontend pousse du code (et inversement).

**paths:**

**- 'frontend/\*\*'**