

# **Synthèse qualDev**

Arthur MALORON  
Steven LEFEBVRE  
Thomas DUCRET

## Table des matières

Table des matières	2
Etat initial du projet	3
Améliorations	4
Coverage :	4
Pipelines et Sonarcloud :	4
Docker :	5
Corrections de bugs :	5
Documentation d'installation :	5
Comment lancer le projet	6

## Etat initial du projet

Initialement, le projet est venu sans documentation d'installation, nous laissant dans le noir quant à la démarche à suivre pour mettre en place le projet.

Pour le lancer, il a donc fallu configurer plusieurs services comme un serveur tomcat ou le déploiement de l'application depuis l'IDE.

L'application était également remplie de bugs, certains étant plus importants que d'autres. Mais on peut par exemple citer le bug qui fait qu'un administrateur n'a aucun moyen de supprimer un utilisateur/un compte ou encore celui qui fait que lorsqu'un administrateur se déconnecte, il puisse encore accéder aux ressources admin du site.

## Améliorations

### Coverage :

Au début du projet, le coverage était d'environ 31%, mais nous avons mis en place de nombreux tests pour augmenter le pourcentage de coverage du projet.

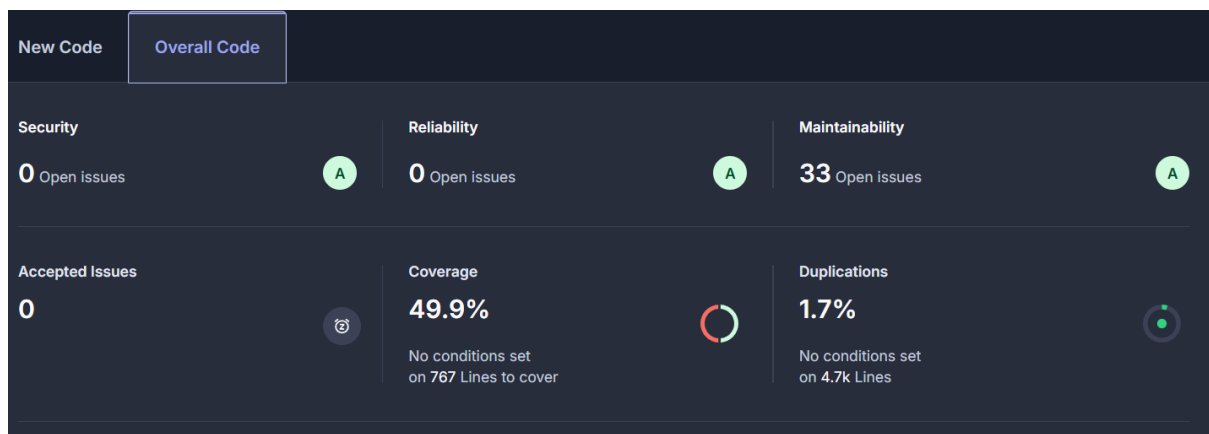
A l'heure de cette synthèse, nous sommes à environ 50% de coverage.

### Pipelines et Sonarcloud :

Au cours du projet, nous avons mis en place une pipeline github responsable de l'analyse automatique du code en utilisant Sonarcloud.

Cette pipeline s'active lors de push sur la branche main et lors de chaque pull request.

Cela permet d'avoir des analyses sur le code global et sur le nouveau code ajouté avec le commit/la pull request afin de garder en tête un objectif de qualité.



Nous avons aussi mis en place une seconde pipeline github servant à tester le déploiement du docker et à run les tests, nous donnant pour chaque fichier de tests quels sont les tests qui ont été passés, lesquels ont échoués, etc.

### *Extrait du résultat :*

```
Tests run: 9, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
-----
Test set: com.iut.banque.modele.TestsClient
-----
Tests run: 26, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.009 sec
-----
Test set: com.iut.banque.modele.TestsCompte
-----
Tests run: 11, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.004 sec
-----
Test set: com.iut.banque.modele.TestsCompteAvecDecouvert
-----
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 sec
-----
Test set: com.iut.banque.modele.TestsCompteSansDecouvert
-----
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
-----
Test set: com.iut.banque.modele.TestsUtilisateur
-----
Tests run: 6, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
-----
Test set: com.iut.banque.utils.TestsHashManager
-----
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec
```

## Docker :

Afin d'accélérer la mise en place du projet et de rendre l'implémentation d'outils dans celui-ci plus simple, nous avons dockérisé le projet.

Le but étant qu'un collaborateur puisse en quelques lignes dans un terminal, lancer le projet.

Les outils que nous avons choisis pour être dockérisés sont :

- Une base de données utilisant mysql:8.0 chargeant les scripts sql afin d'avoir directement une base de données prête.
- Un serveur web HTTP et un jdk en utilisant l'image tomcat:9.0-jdk11
- Maven, qui réalise les tests unitaires en utilisant la base de données dockérisée, mais également compile le projet et l'installe dans le container de Tomcat, le serveur web dockérisé également.

## Corrections de bugs :

Nous avons corrigé plusieurs bugs, bloquants et non-bloquants, afin d'améliorer le fonctionnement de l'application.

Pour en savoir plus, vous pourrez retrouver un rapport au sujet de ces corrections de bugs dans le fichier **Correction\_Bug\_QualDev** dans le dossier Documentation.

## Documentation d'installation :

Nous avons produit une documentation d'installation que vous pourrez retrouver dans le fichier **guide\_installation** dans le dossier Documentation.

Cette documentation pourra vous aider à mettre en place le projet pour le développement sans avoir autant de difficultés que nous, mais si vous voulez simplement déployer le projet ou le tester, vous pouvez vous diriger vers le fichier **guide\_docker** dans le dossier documentation.

## Comment lancer le projet

Pour lancer le projet, le plus simple est de déployer le docker compose qui a été décrit dans la partie [améliorations](#).

Pour obtenir plus de détails sur le lancement du projet en utilisant docker, vous pouvez consulter le document dans le fichier Documentation/guide\_docker.pdf.