Manipuler l’outil de mesure de la qualité du code (SonarQube)

Introduction - Pourquoi la qualité du code ?

La qualité du code assure la **fiabilité**, la **maintenabilité** et la **sécurité** d'un programme. Un mauvais code entraîne des bugs, des coûts élevés et une perte de performance.

Deux types de qualité :

- **Qualité structurelle** : comment une fonctionnalité est implémentée (robustesse, maintenabilité, lisibilité)
- **Qualité fonctionnelle** : le résultat final du logiciel

Les Métriques de Qualité

Définition :

Une métrique est une **caractéristique mesurable** d'un logiciel.

Exemples de métriques

- Nombre de lignes de code
- Nombre de méthodes par classe
- Complexité cyclomatique
- Couplage afférent/efférent
- Niveau d'abstraction et instabilité
- Dette technique

Important : Il n'existe pas de valeur absolue pour une métrique. Les métriques servent à surveiller l'évolution dans le temps et détecter les tendances négatives.

Critères de qualité évalués

- Maintenabilité
- Lisibilité
- Évolutivité
- Robustesse
- Performance

- Sécurité
- Documentation

SonarQube - L'outil d'analyse

Qu'est-ce que c'est ?

SonarQube est un **logiciel open source** qui mesure la qualité du code source. Il analyse automatiquement sans exécuter le code et détecte les bugs, vulnérabilités, mauvaises pratiques et duplications.

- **Langages supportés** : 25+, dont Java, Python, C++, JavaScript, PHP...
- **Analyse** : Code source, design, tests unitaires

Architecture

- **SonarScanner** : exécuteur qui lance les analyses
- **Serveur web** : consultation des résultats via navigateur
- **Base de données** : stockage et historique

Ce que SonarQube Détecte

Rapports et métriques

✓ Densité des commentaires
✓ Couverture des tests unitaires
✓ Respect des conventions de nommage
✓ Respect des règles de codage
✓ Détection de bogues
✓ Détection de code mort
✓ Détection de code dupliqué
✓ Complexité et complexité cognitive
✓ Scores de maintenabilité, fiabilité et sécurité
✓ Dette technique

Les 7 axes de qualité

1. Architecture & design

2. Documentation

3. Respect des standards de codage
4. Non-duplication du code
5. Tests unitaires
6. Complexité
7. Bogues potentiels

Classification des Défauts

SonarQube classe les défauts en trois catégories :

Type	Description	Impact
Bugs	Erreurs dans le code qui provoquent un mauvais fonctionnement de l'application	Fiabilité
Vulnérabilités	Faiblesses du code exploitables par des attaquants	Sécurité
Code smells	Mauvaises pratiques dans le code (code dupliqué, répété, commentaires inutiles)	Maintenabilité

Tableau de Bord SonarQube

Le tableau de bord affiche :

- Scores de maintenabilité, fiabilité et sécurité

Score	Fiabilité	Sécurité	Maintenabilité
A	0	0	ratio dette technique ≤ 5%
B	≥ 1 mineur	≥ 1 mineure	6% ≤ ratio dette technique ≤ 10%
C	≥1 majeur	≥ 1 majeure	11% ≤ ratio dette technique ≤ 20%
D	≥ 1 critique	≥ 1 critique	21% ≤ ratio dette technique ≤ 50%
E	≥ 1 bloquant	≥1 bloquante	ratio dette technique ≥ 51%
Issue	Bug	Vulnerability	Code smell

- Taux de couverture des tests
- Taux de duplication
- Taille du projet
- Langages utilisés

Les tests unitaires

Les tests unitaires sont des tests qui consistent à vérifier le bon fonctionnement d'une petite partie isolée d'un programme (fonction, méthode, objet). Ils sont largement utilisés dans les méthodes agiles.

Ils permettent de :

- vérifier que la logique du code est correcte dans tous les cas ;
- faciliter la compréhension et la lisibilité du code ;
- servir de documentation du projet ;
- détecter rapidement les erreurs grâce à des tests rapides (en millisecondes) ;
- sécuriser les modifications futures du code (refactoring) ;
- améliorer la qualité globale du logiciel.

Le test unitaire permet aussi de travailler sur des parties du projet indépendamment des autres.

En PHP, PHPUnit est un framework open source dédié aux tests unitaires.
Il permet notamment de faire des tests de régression à l'aide d'assertions, et peut être installé facilement via Composer.

Exemples de questions d'EFM :

8. Qu'est-ce que le SonarQube? **(2 pts)**

a. Une pratique d'ingénierie logicielle qui vise à réunir l'équipe de développement et l'équipe d'exploitation dans le but d'automatiser le projet à chaque étape.

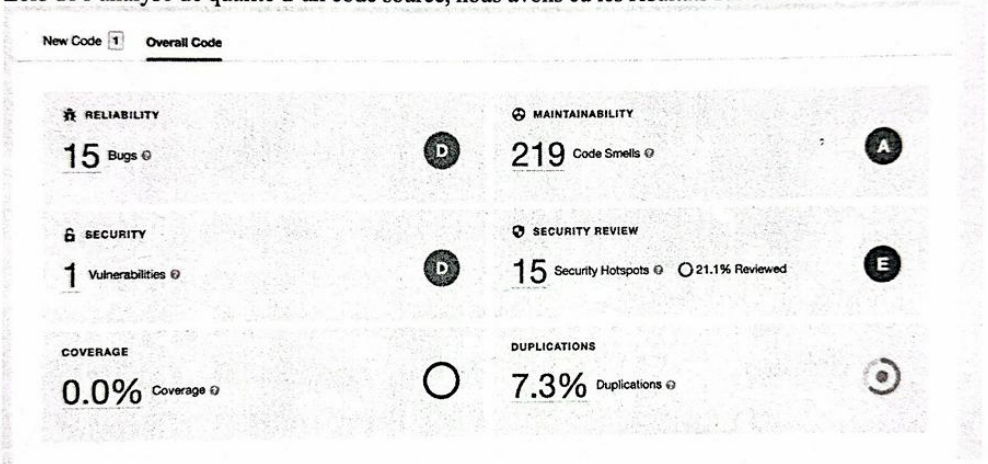
b. est un logiciel open source de mesure de la qualité du code source de projets de développement.

c. Est une méthode de gestion de projet qui applique les principes du développement agile.

2. Quels types de problèmes de qualité de code peut détecter SonarQube ? (2pts)

Dossier3 (6pts)

Lors de l'analyse de qualité d'un code source, nous avons eu les résultats suivants :



1. Quel est le nom du logiciel utilisé ? 2pts
2. Que ségnifie la lettre « A » dans MANTENABILITY ? 2pts
3. Quelle est la différence entre un Bug et un « Code Smell » ? 2pts