

Année universitaire : 2023/2024

Conception et réalisation d'une application pour la gestion de pour de soutien

Ingénierie Informatique et Réseaux

Réalisé par :

Adnane Elayachi, Hamza Zerouali, Yahya
Elatare

Table de matières

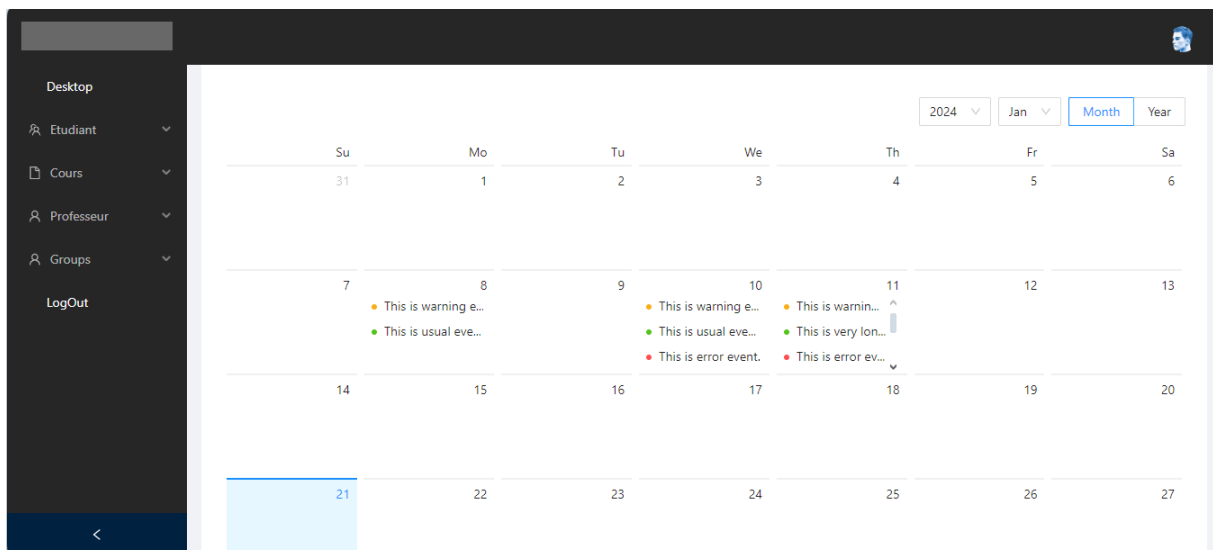
1. Introduction
2. Architecture Microservices
3. Conception des Microservices .
4. Conteneurisation avec Docker...
5. CI/CD avec Jenkins
6. Déploiement Automatique .
7. Intégration de SonarQube .
8. Conclusion .

Introduction

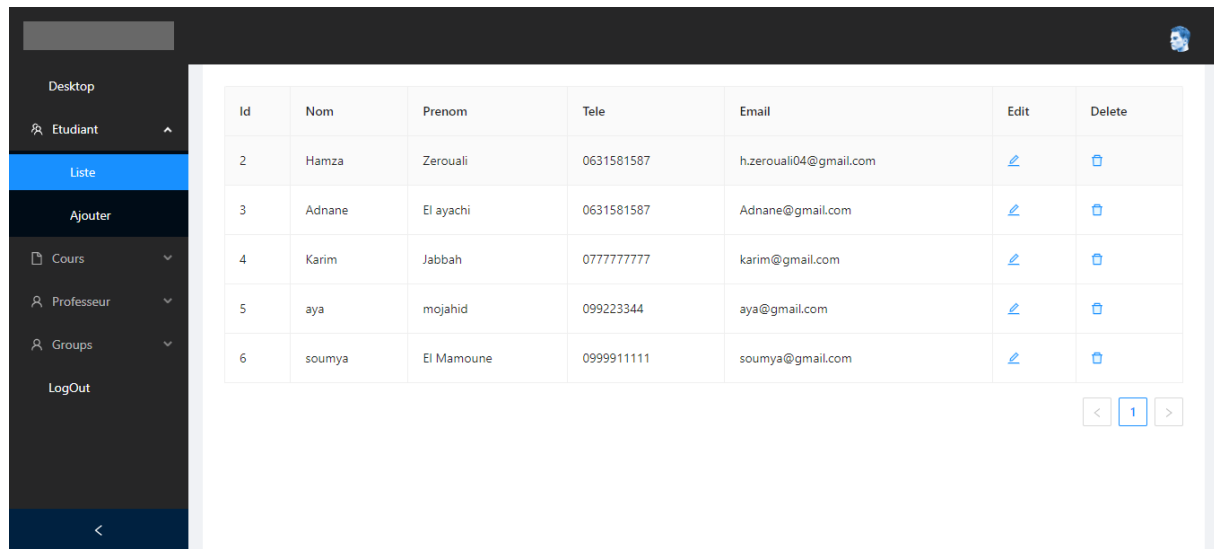
Notre projet de gestion de cours de soutien repose sur une architecture micro-services, soulignant l'importance de cette approche moderne dans le domaine du développement logiciel. Composée de quatre modules distincts dédiés aux étudiants, professeurs, cours et groupes, cette architecture offre une modularité exceptionnelle. Chaque micro-service, développé avec le Framework Spring et soutenu par une base de données MySQL, est conçu pour permettre une évolutivité aisée et une maintenance simplifiée. L'utilisation de Jenkins pour l'intégration continue et de SonarQube pour garantir la qualité du code renforce notre engagement envers les meilleures pratiques de développement. Cette stratégie innovante vise à simplifier la gestion des cours de soutien en offrant une flexibilité accrue pour répondre aux besoins variés des utilisateurs, tout en assurant une performance optimale et une facilité de maintenance. L'adoption de l'architecture micro-services réaffirme ainsi notre engagement envers des solutions modernes et agiles dans le développement de logiciels.

Interfaces (Angular) :

- Planning Cours :



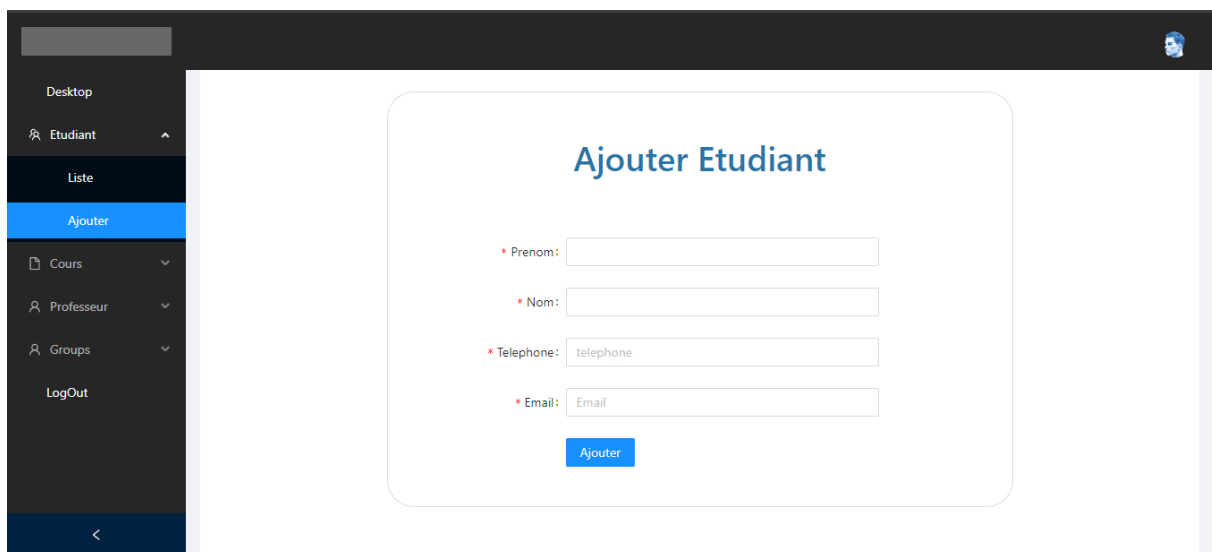
- Liste Etudiant :



Id	Nom	Prenom	Tele	Email	Edit	Delete
2	Hamza	Zerouali	0631581587	h.zerouali04@gmail.com		
3	Adnane	El ayachi	0631581587	Adnane@gmail.com		
4	Karim	Jabbah	0777777777	karim@gmail.com		
5	aya	mojahid	099223344	aya@gmail.com		
6	soumya	El Mamoune	0999911111	soumya@gmail.com		

< 1 >

- Ajout d'Etudiant



Ajouter Etudiant

* Prenom:

* Nom:

* Telephone:

* Email:

- Liste Matiers :

localhost4200/dashbord/listC

Applications hamza stage Dr. Driss ESSABBAR Pricing Learn C Programming Exercices en langag... Pointers and Refere... mailleur site

Desktop

Etudiant

Cours

List

Ajouter

Professeur

Groups

LogOut

Id	Titre	Description	Edit	Delete
2	Math	bac2		
8	j2ee	informatique		

< 1 >

- Liste groupe :

localhost4200/dashbord/groups

Applications hamza stage Dr. Driss ESSABBAR Pricing Learn C Programming Exercices en langag... Pointers and Refere... mailleur site

Desktop

Etudiant

Cours

List

Ajouter

Professeur

Groups

List

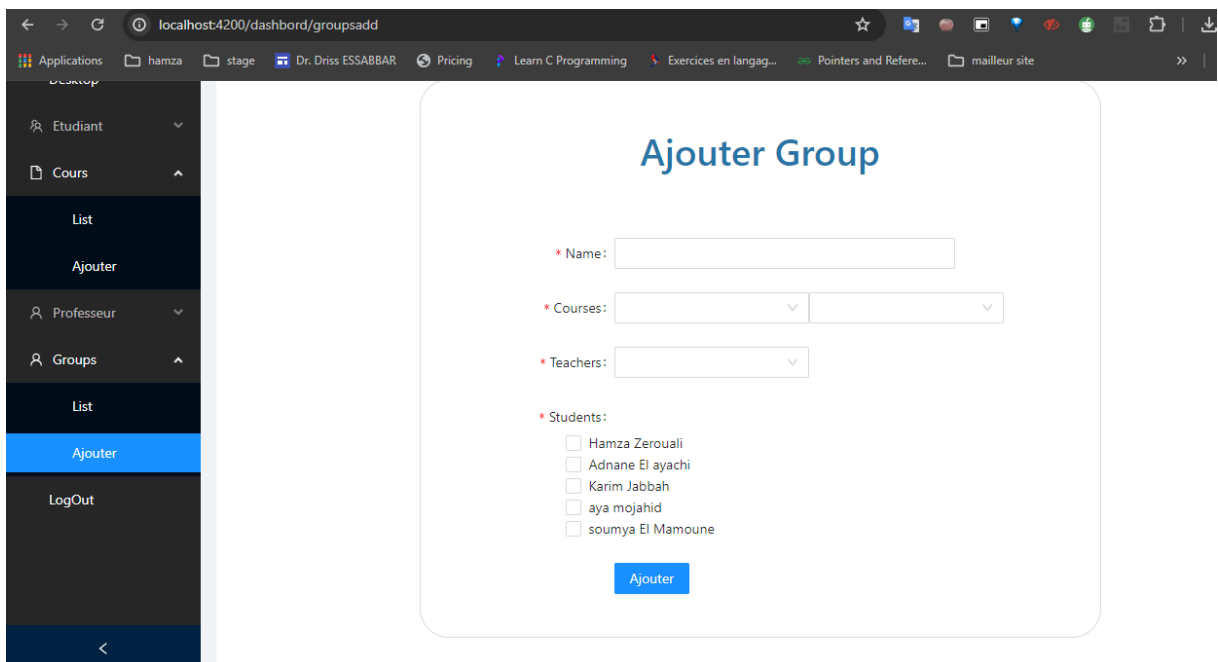
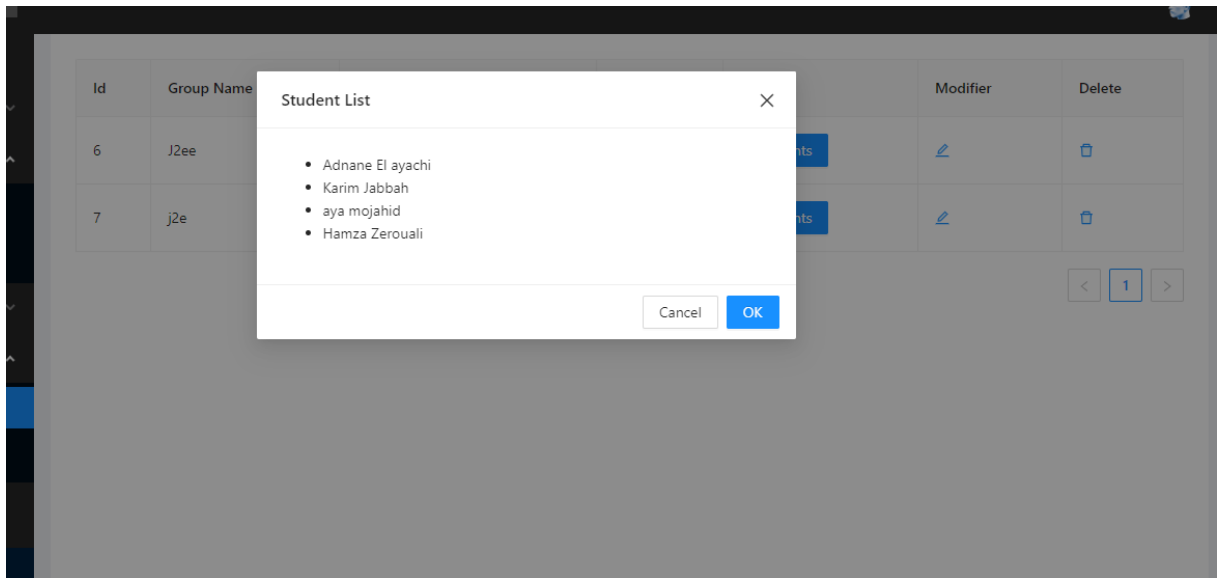
Ajouter

LogOut

Id	Group Name	Teacher	Course	Etudiant	Modifier	Delete
6	J2ee	Soumya El Mamoun	Math	Students		
7	j2e	Soumya El Mamoun	j2ee	Students		

< 1 >

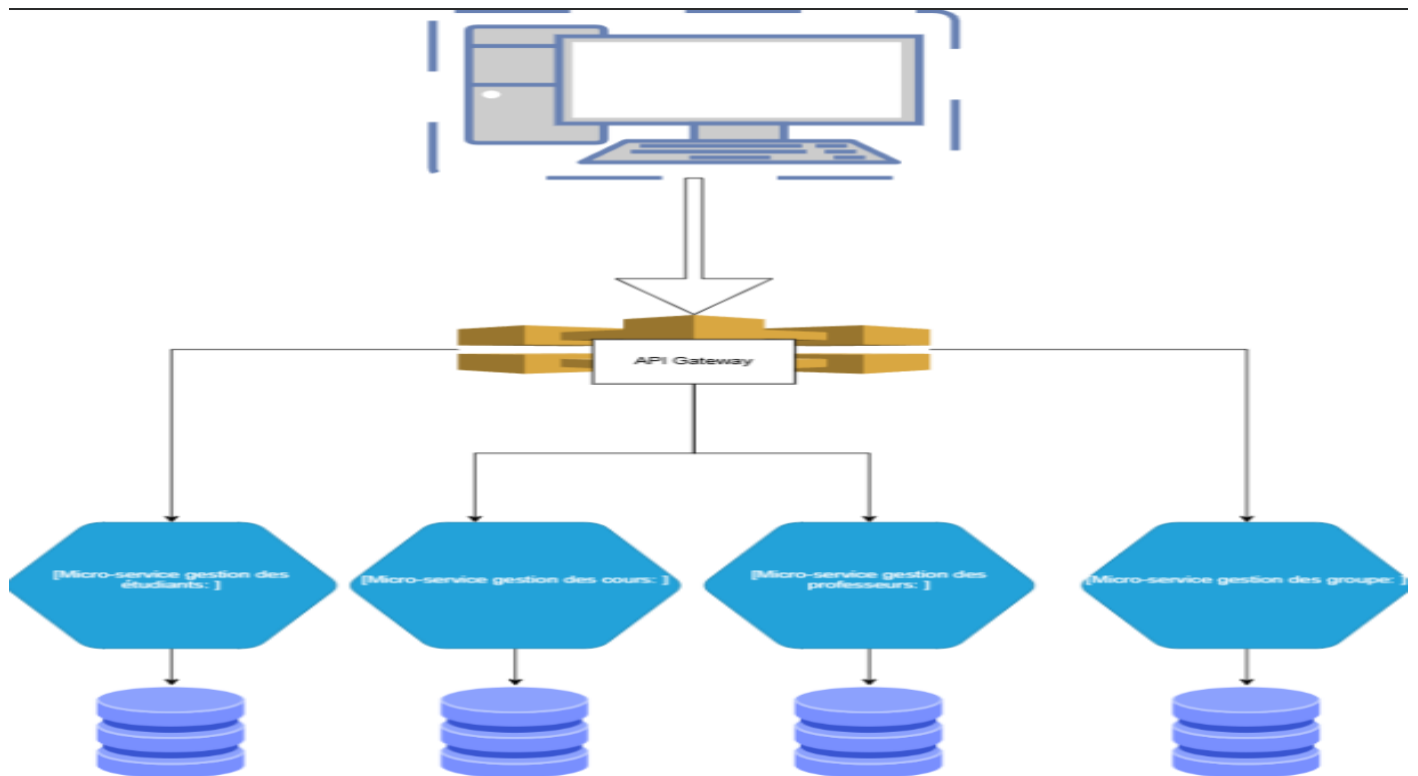
- Etudiant dans cours :



Architecture Micro-services

Notre architecture micro-services pour la gestion des cours de soutien comprend quatre micro-services dédiés (étudiants, professeurs, cours, groupes) avec des bases de données indépendantes. Nous avons intégré Consul pour la découverte de services et mis en place une passerelle avec RestTemplate pour faciliter la communication entre les micro-services. Cette approche modulaire offre flexibilité, évolutivité et maintenance simplifiée. Consul permet la découverte automatique des services, tandis que la passerelle assure un point d'entrée unique pour les clients. RestTemplate simplifie les appels entre micro-services, favorisant une communication fluide et dynamique. Globalement, notre architecture optimise la modularité et la performance du système.

- Architecture :



Conception des Micro-services

Notre conception des micro-services pour la gestion des cours de soutien repose sur une approche modulaire et spécialisée, avec quatre micro-services distincts, chacun dédié à des aspects spécifiques du système.

Micro-service Étudiant :

- Gère les informations relatives aux étudiants, incluant les données personnelles, résultats académiques et détails de contact.
- Permet l'inscription, la mise à jour et la suppression d'étudiants.
- Offre des fonctionnalités de recherche et de filtrage pour accéder rapidement aux informations spécifiques des étudiants.

Micro-service Professeur :

- Gère les détails des professeurs, y compris compétences, horaires de disponibilité et historique des cours dispensés.
- Permet l'ajout, la modification et la suppression des professeurs.
- Fournit des fonctionnalités avancées telles que la recherche par compétences.

Micro-service Cours

- Centralise les informations sur les cours de soutien, incluant les matières, niveaux académiques, horaires et lieux.
- Permet d'ajouter, de mettre à jour et de supprimer des cours.
- Facilite la recherche et la planification des cours en fonction des critères spécifiques.

Micro-service Groupe :


- Gère la formation et la gestion des groupes d'étudiants pour les cours collectifs.
- Permet de créer, de modifier et de dissoudre des groupes.
- Offre des fonctionnalités de recherche pour simplifier la gestion des groupes.

Chaque micro-service fonctionne de manière autonome avec sa propre base de données, favorisant ainsi l'indépendance et la modularité. La communication entre ces micro-services est orchestrée à l'aide de Consul pour la découverte de services et une passerelle avec RestTemplate pour une communication fluide. Cette approche offre une flexibilité accrue, une maintenance simplifiée et une

évolutivité optimale pour répondre aux besoins spécifiques de chaque composant du système de gestion des cours de soutien.

- Description des services

Services <small>6 total</small>	
<input type="text" value="Search"/>	<div>Search Across ▼Health Status ▼Service Type ▼</div>
<div><div>✖ Gateway</div><div>1 instance</div></div>	
<div><div>✖ SERVICE-GROUPEC</div><div>1 instance</div></div>	
<div><div>✖ SERVICE-PROFESSEUR</div><div>1 instance</div></div>	
<div><div>✔ consul</div><div>1 instance</div></div>	
<div><div>✔ SERVICE-COURS</div><div>1 instance</div></div>	
<div><div>✔ SERVICE-ETUDIANT</div><div>1 instance</div></div>	

▼  Spring Boot
▶ CoursApplication [devtools] :8082/
▶ EtudiantApplication [devtools] :8081/
▶ GatewayApplication :8888/
▶ GroupeApplication [devtools] :8083/
▶ ProfesseurApplication [devtools] :8084/

- Mécanismes de communication

RestTemplate, est un composant de Spring Framework conçu pour simplifier les appels RESTful entre applications Java. L'utilisation de RestTemplate avec Consul facilite la communication entre microservices en simplifiant la découverte des services et en offrant des fonctionnalités pratiques pour effectuer des appels HTTP.

Cela contribue à rendre l'architecture de microservices plus flexible et résiliente.

```
@BushidoZr <h.zerouali04@gmail.com>
@Bean
public RestTemplate restTemplate() {
    RestTemplate restTemplate = new RestTemplate();
    SimpleClientHttpRequestFactory requestFactory = new SimpleClientHttpRequestFactory();
    requestFactory.setConnectTimeout(5000);
    requestFactory.setReadTimeout(5000);
    restTemplate.setRequestFactory(requestFactory);

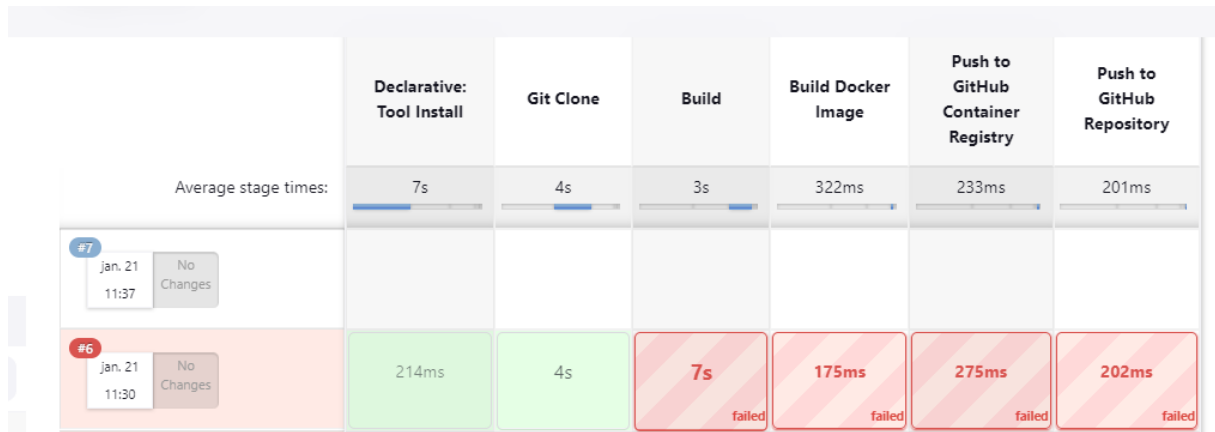
    return restTemplate;
}
```

Conteneurisation avec Docker

Docker offre une solution puissante pour l'implémentation d'applications dans des conteneurs, offrant des avantages significatifs. Sa portabilité assure une adaptabilité maximale entre les environnements de développement, de test et de production. L'isolation des conteneurs évite les conflits de dépendances, assurant cohérence et stabilité. La facilité d'évolutivité permet une gestion flexible des charges de travail. La rapidité de déploiement réduit le temps nécessaire par rapport aux méthodes traditionnelles. La gestion des versions, facilitée par les concepts d'images et de registres, garantit la cohérence des déploiements. Les outils tels que Docker Compose simplifient la configuration et la coordination d'applications complexes, renforçant ainsi la facilité de gestion globale.

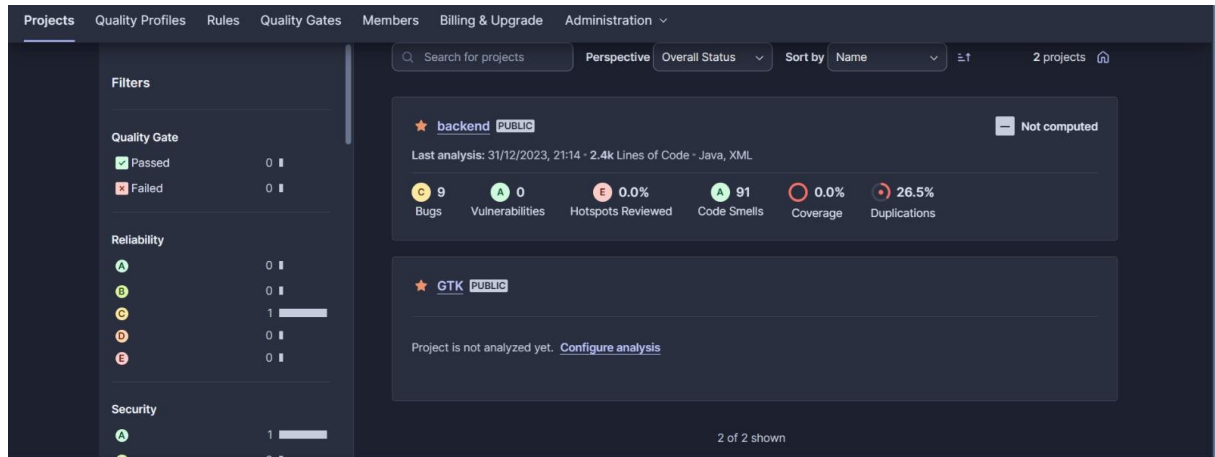
		gcs_mic	Running (6/8)	N/A	N/A	N/A	N/A			
		mysql bf1333	mysql:latest Running	N/A	N/A	N/A	N/A			
		consu 64a291	consul:1.15 Running	N/A	N/A	N/A	N/A			
		phpm 88515t	phpmyadm Running	N/A	N/A	N/A	N/A			
		gatew 549e2t	gcs_micros Running	N/A	N/A	N/A	N/A			

CI/CD avec Jenkins



```
1 pipeline {
2   agent any
3
4   tools {
5     maven 'maven'
6   }
7
8
9
10  stages {
11    stage {
12      script {
13        checkout([$class: 'GitSCM', branches: [[main: 'main']],
14          userRemoteConfigs: [[url:
15            'https://github.com/Hamzazr/gcs_MicroS.git']]])
16      }
17    }
18  }
19
20  stage('Build') {
21    steps {
22      bat 'mvn clean install'
23    }
24  }
25
26  stage('Build Docker Image') {
27    steps {
28      script {
29        def dockerImage = docker.build("h.zerouali04@gmail.com/app_gcs:${env.BUILD_NUMBER}")
30      }
31    }
32  }
33 }
```

Intégration de SonarQube



Conclusion

Le projet consiste en une plateforme web dédiée à la gestion des cours de soutien, Elle offrant la possibilité aux administrateurs de gérer les inscriptions, les absences, les abonnements, la planification des cours pour les professeurs et les tuteurs, Les enseignants ont la capacité de gérer leurs plannings et de suivre l'évolution de leurs étudiants. Nous sommes convaincus que cette plateforme a le potentiel de révolutionner la gestion des cours de soutien.