



Objectifs

- Réaliser l'itération 6 de l'application Gestion Log
 - Envoi des informations de log SSH sur un serveur centralisé
 - Valider les exigences du CDC (REQ005 ET REQ006)

Contenu technique

- Utilisation d'une API REST
- Formatage Json

Durée 5 heures

I. Vérification de l'API POST du serveur centralisé

Le serveur centralisé de log a été installé sur la VM « Serveur SSH »

- Q1. Donner la liste des routes définies dans l'API
- Q2. Donner la définition et le rôle d'une requête HTTP GET.
- Q3. Donner la définition et le rôle d'une requête HTTP POST
- Q4. En utilisant Firefox, tester les différentes routes du serveur.
- Q5. En utilisant le plugin firefox RESTED ou postman, tester les 3 routes utilisant le protocole HTTP GET
- Q6. Donner les éléments de chaque entrée de log
- Q7. En utilisant le plugin firefox RESTED, tester la route utilisant le protocole HTTP POST (pour celle-ci n'oubliez pas de préciser le **Content-Type application/json** dans le Header)



II. Utiliser Poco

Q8. Rechercher le rôle de la librairie C++ POCO

II.1 Dépôt PocoTest

Un exemple d'utilisation de la librairie Poco est proposée sur ce dépôt

https://gitlab.sn-kastler.fr/ciel/depots_gestion_centralis-_de_logs/testpoco.git

Q9. Cloner le dépôt et ouvrir le projet

II.1.1 Fonction PocoGet()

Q10. Étudier la fonction PocoGet()

Q11. Quel est son rôle ?

Q12. Donner la définition d'une URI

Q13. Que réalise la méthode Poco::Net::HTTPRequest ? Est ce une requête GET ou POST ?

Q14. Que réalise la méthode HTTPClientSession ?

Q15. Tester la méthode pour afficher le contenu de la route /api/v1/sshlog/all

Q16. Tester la méthode pour afficher le contenu de la route /api/v1/sshlog?id=x

II.1.2 Fonction PocoPost()

Q17. Étudier la fonction PocoPost()

Q18. Quel est son rôle ?

Q19. Que réalise la méthode Poco::Net::HTTPRequest ? Est ce une requête GET ou POST ?

Q20. Quel est le rôle de Poco::JSON::Object obj ?

Q21. Grâce à quelle méthode de Poco::JSON::Object on peut ajouter des éléments à Poco::JSON::Object ?



III. Intégration de la librairie POCO à votre projet

III.1 Git

N'oubliez pas de vous replacer sur la branche de dev pour cette itération.

III.2 Ajout de la librairie Poco

Étudier le `CMakeList.txt` et le fichier `conanfile` du projet de test pour intégrer la librairie Poco à votre projet

III.3 PocoGetAll()

Q22. En vous référant au DDC, ajouter la fonction `PocoGetAll()` à votre projet.

III.4 PocoGetLogId()

Q23. En vous référant au DDC, ajouter la fonction `pocoGetLogId()` à votre projet.

III.5 PocoPostSshLog()

Objectif : La fonction `PocoPostSshLog()` peut être appelée soit :

- lors de la visualisation locale des logs ssh
- soit via le menu pour envoyer sur le serveur les logs enregistrés dans le fichier `ssh_connexion.txt`

Dans les deux cas, les logs sont automatiquement envoyés sur le serveur centralisé

Q24. En vous référant au DDC, ajouter la fonction `PocoPostSshLog()` à votre projet pour lors de la visualisation locale des logs ssh

III.6 Validation

Appeler l'enseignant pour qu'il valide votre travail



IV. Dossier de validation

Vérifier que votre application répond aux exigences du cahier des charges.

Q25. Compléter le dossier de vérification Gestion_Log_DDV.odt (ESS004)

V. Outils DevOps

Afin de finaliser votre travail, n'oubliez pas :

- D'indenter correctement votre code
- De documenter le code et les fonctions

VI. Livrable

Sur le moodle de la section

- Votre compte rendu de TP avec les réponses aux questions de celui-ci
- La documentation générée (Télécharger l'artifact sur le pipeline)
- Le dossier de validation DDV

Sur le serveur GIT de la section

- Le projet Gestion_Log Branche dev
- Merge de la branche dev sur master
- Le tag de la version v6.0

VII. Améliorations

Grâce à ce projet, vous avez créé une application qui permet de lister, filtrer et enregistrer des logs importants sur votre PC et également de les transmettre à un serveur centralisé.

Cependant, cette application n'est pas parfaite.

Pour conclure ce projet, le RSSI vous demande un retour d'expérience pour améliorer son produit.

Q26. Critiquer l'application notamment en listant les pistes d'améliorations