

À Spécifier :

- *Temps passé par tâches*
- *Justification*
- *Résultat obtenu*
- *Bilan/Avancé sur le projet*
- *Planification Séance d'après*

SPRINT 2

27/03/2024

Yao:

- Recherche sur les méthodes pour obtenir la consommation énergétique des pipelines jenkins visualisés avec Graphana
- Résultats: On ne peut visualiser la consommation des pipelines en les mettant dans des cgroups ou dans des dockers
- Suite: Je recherche actuellement à identifier les pipelines sur graphana afin de pouvoir récupérer les informations des pipelines en particulier

Moussa:

- Ajout des des fonctionnalités du sprint 1 dans le plugin
- Resultat: j'ai configuré l'environnement maven qui présentait de gros bugs. Cela m'a pris assez de temps. Avec l'aide du prof (suppression du fichier .m2) j'ai résolu le problème. Ensuite j'ai commencé l'intégration des fonctionnalités. Mais celle-ci ne marchait pas correctement.
- Cause : Je mettais seulement le code de la pipeline jenkins sans la lier à une classe java.
- Temps:

Axel:

Configuration de l'environnement du Plugin

Test sur des projets Github

Début de l'implémentation des fonctionnalité précédentes sous le format de plugin

Nous avons rencontré un problème, cette implémentation est en réalité beaucoup plus compliquée que ce que nous prévoyons, nous pensions pouvoir utiliser notre script groovy hors cela ne fonctionne pas ainsi On va donc beaucoup dépassé le temps que l'on prévoyait pour cette tâche.

La tâche de configuration a pu rentrer à peu près dans les temps (2h à la place de 1h), il n'y a rien eu d'inattendu.

09/04/2024

Yao:

- Prévu: Identification des build
- Résultats: On ne peut visualiser la consommation des pipelines qu'en les mettant dans des cgroups ou dans des dockers
- Suite: Je recherche actuellement à identifier les pipelines sur grafana afin de pouvoir récupérer les informations des pipelines en particulier

Moussa:

- prévu: Terminer l'intégration des fonctionnalités
- Résultats: J'ai commencé l'intégration avec Axel puis par la suite il a continué sur l'intégration et j'ai continué sur la mise en place d'une UI pour l'affichage du graph de consommation
- Suite: Je recherche actuellement des templates html et js pour la mise en forme du graphique

Axel:

Poursuite de l'intégration de la mesure de consommation dans le plugin. Plusieurs méthodes ont été envisagées, ce qui a causé certaines pertes de temps. Un peu de temps passé à aider Moussa sur la partie UI.

La fonctionnalité a finalement bien pu être implémentée correctement.

Grâce à ce qui avait été fait la séance précédente l'adaptation pour tout type de projet a pu être terminée.

30/04/2024

Ce jour a été à cheval entre le sprint 2 et 3

Yao : Recherche sur un bug (AddressSanitizer Lié au Lancement de PowerAPI détecté par le capteur HWPC)

Lorsqu'on lance PowerAPI, on obtient un Bug :AddressSanitizer|DEADLYSIGNAL détecté par le senseur HWPC. A ce stade, pas encore trouvé la raison du bug. Cela a grandement ralenti l'avancée du travail

Moussa : Avancer sur la documentation des tâches effectuées. Commencement des recherches sur le sauvegarde dans une base de données influxdb. J'ai rencontré des problèmes d'identifiant pour la connexion à la base de données.

Axel : Test sur machine Linux avec l'affichage en histogramme de la consommation du build; Récupération de la conso des builds précédents;

SPRINT 3

17/05/2024

Yao : Recherche sur le bug AddressSanitizer. J'ai aussi rencontré un autre Bug Lors du lancement de PowerAPI au niveau de docker qui m'empêchais de stocker mes images et mes containers sur la machine. Mais ce bug était plus lié à la machine qu'à PowerAPI ce qui m'a fait perdre assez de temps.

Aussi le Bug AddressSanitizer me ralentit toujours.

Moussa :

- Réussir l'installation de la base de données influxdb et mise en place d'un programme java capable de faire des requêtes en direction de celle-ci (read/write).
- Durant cette étape, j'ai rencontré des problèmes d'identifiants pour une première connexion à influxdb
- Avec la documentation j'ai pu résoudre le problème.

Axel : Affichage de l'historique de consommation des builds, une erreur due à une mauvaise interprétation d'un String par JavaScript a causé une erreur qui a pris du temps à être résolue. Mais finalement la fonctionnalité a bien pu être implémentée. (De nouvelles classes ont dû être mises en place pour le transfert des variables entre Java - Jenkins - JavaScript).

24/05/2024

Yao : Intégration des données pour chaque stage des builds

Moussa : Création de deux projets test pour vérifier le fonctionnement du plugin que j'ai pushé sur deux branches d'un dépôt public github. Puis à l'aide d'un code pipeline contenant plusieurs étapes, j'ai créé un projet pipeline sur jenkins appelant les dépôts git pour observer la consommation.

Axel :

- Entre les séances : mise en place server pour installer le plugin par URL plutôt que par dépôt de fichier. Ajout d'un formulaire pour que l'utilisateur puisse choisir le nombre de builds antérieur qu'il veut afficher. Essais en exécutant les builds via un autre agent (VM ubuntu) via ssh. Essais de mesure de consommations sur Mac OS. Création d'un nouveau listener permettant une lecture séparée par Stages (Nodes d'exécution Jenkins) avec affichage dans la console des temps pour de début pour chacun
- (Création d'un nouveau listener pour cette fonctionnalité constituant un thread tournant en parallèle de celui déjà fonctionnelle précédemment)
- Séance : Intégration graphique de la consommation répartie par stage