

Rapport de tests de charge

Load testing de l'application StackOverflow avec JMeter

AMT-Long-du-Zboub

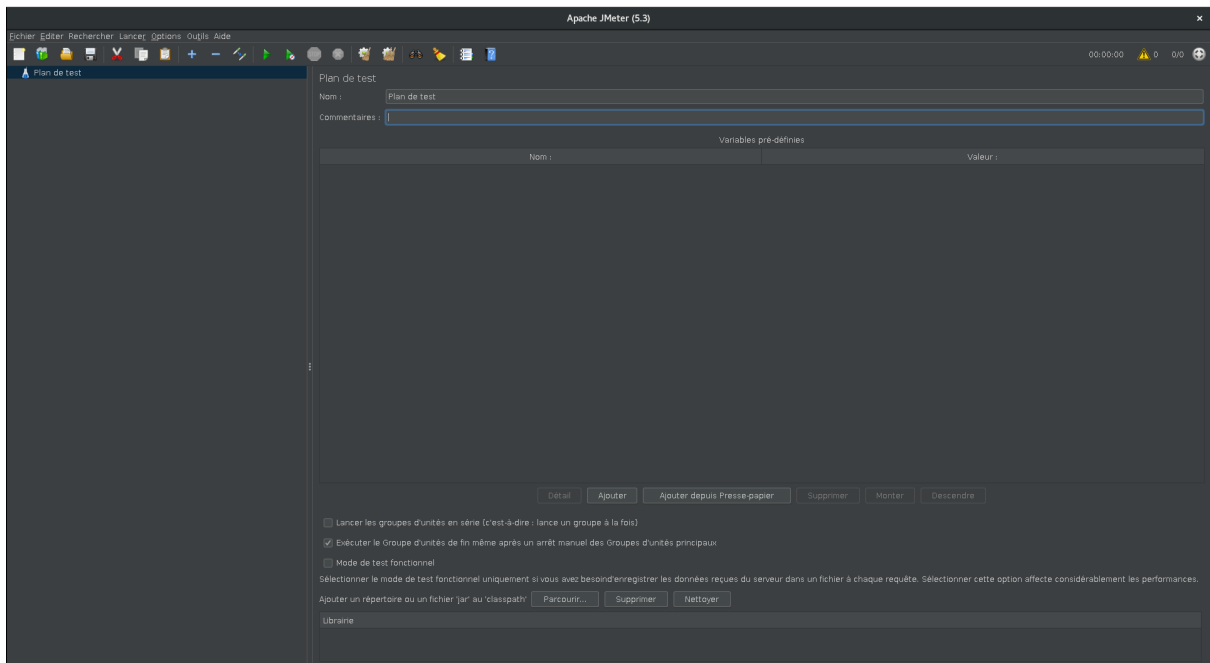


Figure 1: Boîte de dialogue JMeter.

Laboratoire d'Application Multi-Tiers (AMT)

Prof. Olivier Liechti

Assist. Daniel Palumbo



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

October 30, 2020

Table des matières

Table des matières	1
1 Introduction	2
1.1 Introduction	2
1.2 Pourquoi un rapport de test JMeter ?	2
2 Les tests effectués	2
2.1 Présentation des tests	2
2.2 Scénario 1: Utilisation générale du site	2
2.2.1 Objectif du test	2
2.2.2 Scénario de test	3
2.2.3 Résultats obtenus	3
2.2.4 Interprétation des résultats	3
2.3 Scénario 2: Prise en compte des votes	3
2.3.1 Objectif du test	3
2.3.2 Scénario de test	3
2.3.3 Résultats obtenus	4
2.3.4 Interprétation des résultats	4
3 Conclusion	4

1 Introduction

1.1 Introduction

Dans le cadre du projet de conception d'une application StackOverflow, il nous était demandé d'effectuer différents types de tests de notre application. Pour effectuer les tests de charge de notre application, nous avons utilisé l'outil JMeter. JMeter permet de simuler plusieurs utilisateurs de notre application (via différents threads) et nous permet ainsi de tester en local l'utilisation de notre application en simultanés par plusieurs utilisateurs. JMeter crée des requêtes et les envoie au serveur d'application comme un browser et lorsqu'il reçoit des réponses du serveur, il peut ainsi les traiter, en faire des charts etc ...

1.2 Pourquoi un rapport de test JMeter ?

Les tests unitaires jUnit ou e2e permettent de tester le fonctionnement de l'application. Et, le simple fait que les tests passent ou ne passent pas sont un indicateur suffisant pour montrer si l'application marche ou ne marche pas. L'utilisation de ces tests est relativement binaire. Or, dans le cas de tests de charge, il ne s'agit pas de voir si l'application marche ou ne marche pas mais de recueillir des données d'utilisations de l'application telles que par exemple:

- Quelle est en moyenne le temps de réponse d'une page lorsque 500 utilisateurs se connectent au site
- Ou encore au bout de combien de connexions de différents utilisateurs, le site crashe, etc.

Le fait donc de présenter les résultats des tests dans un rapport paraît donc plutôt pertinents dans le cas de tests de charge.

2 Les tests effectués

2.1 Présentation des tests

Pour faire les tests JMeter, on utilise la méthode suivante. On choisit quelques scénarios d'utilisation de notre application que nous soumettons à des charges plus ou moins lourdes (il y a un nombre d'utilisateurs plus ou moins élevés) pour voir comment réagit l'application lorsqu'elle est soumise à ces différentes charges.

Pour chaque requêtes HTTP effectués sur le site, on s'assure de son succès à l'aide d'une assertion et on affiche les résultats à l'aide d'un arbre de résultat et un rapport consolidé.

On utilise un gestionnaire de cookies pour gérer les sessions des différents utilisateurs

Un fichier csv nous permet d'assurer l'enregistrement de 100 personnes(ou 500) avec des champs préremplis pour l'enregistrement. De même pour la connexion au site.

En ce qui concerne les réponses ou les votes, on a besoin de récupérer les bons URL (qui change en fonction de l'id de la question) et pour ça, on peut utiliser l'extracteur d'expressions régulières.

2.2 Scénario 1: Utilisation générale du site

2.2.1 Objectif du test

Le premier test effectué est un test d'utilisation générale du site. On veut pouvoir tester différentes charges sur le site de manière réaliste et voir comment réagit le site web à ces différentes charges. Pour cela on va utiliser deux charges différentes : une première fois avec 100 utilisateurs connectés sur le site et une autre fois avec 500 utilisateurs connectés sur le site.

2.2.2 Scénario de test

Pour ce test ci, chaque utilisateur qui se connecte sur le site respectera le scénario suivant:

- L'utilisateur crée un compte
- L'utilisateur se connecte
- L'utilisateur ajoute une question
- L'utilisateur va dans la liste des questions
- L'utilisateur répond à la question
- L'utilisateur commente la question
- L'utilisateur upvote la réponse

Dans le cas de la première charge à tester de 100 utilisateurs:

Pour tester de manière plus réaliste, on spécifie dans le groupe d'unités une durée de montée en charge de 500 secondes. De cette manière toutes les 5 secondes, 1 utilisateur différent se connecte au site.

Dans le cas de la deuxième charge à tester de 500 utilisateurs:

La montée en charges sera de 1500 secondes (1 utilisateurs/ 3 secondes)

2.2.3 Résultats obtenus

Plan de test General.jmx contient le test à visualiser

2.2.4 Interprétation des résultats

On s'attend à voir que la réactivité du site est légèrement impactée par le nombre d'utilisateurs mais cela dans une mesure acceptable. Dans le cas où nous aurions monté un cahier des charges précis avec des attentes sur la réactivité du site nous aurions pu confronter nos attentes avec les résultats JMeter. Mais dans le cadre de ce projet, les résultats que nous avons obtenus sont satisfaisants.

2.3 Scénario 2: Prise en compte des votes

2.3.1 Objectif du test

Le test 2 est un test plus spécifique que le test 1 qui se concentre sur la feature des votes. L'objectif de ce test est de voir les votes sont correctement pris en compte dans le cas où un certain nombre d'utilisateurs se connectent sur le site et votent pour une question en particulier.

2.3.2 Scénario de test

Un seul utilisateur effectue ceci:

- L'utilisateur crée un compte
- L'utilisateur se connecte
- L'utilisateur ajoute une question
- L'utilisateur va dans la liste des questions
- L'utilisateur répond à la question

100 (ou 500 en fonction de la charge à tester) effectuent ceci:

- L'utilisateur crée un compte
- L'utilisateur se connecte
- L'utilisateur va dans la question
- L'utilisateur upvote la réponse

2.3.3 Résultats obtenus

Plan de test Vote.jmx contient le test à visualiser.

2.3.4 Interprétation des résultats

On s'attend à voir que le résultat obtenu est bien celui que nous attendions c'est à dire que le nombre de upvote effectué par chaque threads est le nombre indiqué par la page de question et ce quelque soit le load (100 ou 500 utilisateurs)

3 Conclusion

En conclusion, on peut déjà dire que étant donné que les tests sont effectué en local, il est possible que les résultats des tests soient limités par les capacités de ma propre machine et pas forcément par l'application en elle même. Parfois, en cas de bugs présents sur ma machine, il était difficile d'exécuter les tests JMeter puisque ce logiciel agit exactement de la même manière que le browser. Ensuite, il y aurait peut être d'autres scénarios intéressants à tester mais en ce qui concerne notre application et les features que nous avons pu implémenter, d'autres scénarios ne semblaient pas forcément pertinents à tester.