

Plan de test de performance

Défis tes copains (Projet Annuel)

I - Description de l'application

a - Description global du projet

Le site défis tes copains , est un site communautaire , et de réseau social créé dans le but du projet annuel de la 4ème année. Le site propose la création de Challenge afin que les utilisateurs puissent s'inscrire et partager sur celui-ci . De plus un système de groupe et d'ami à été implémenté sur le site afin de faciliter la discussion entre les utilisateurs. Le site propose une partie premium pour les entreprises ou école payante , afin de créer des groupes et challenges restreint à certaines personnes, et avec un nombre plus élevé de participants.

b - Objectif de l'application

L'objectif du site Défis tes copains est de rassembler des utilisateurs autour de challenge , et de permettre le partage sur des expériences vécues à travers ceci grâce à un espace Post et Commentaire sous chaque challenge. Les utilisateurs pourront créer des groupes pour participer à des challenges entre eux. De plus, un système d'ami et de messagerie sera implémenté pour permettre la communication au sein du site.

c - Type d'utilisateurs prévus

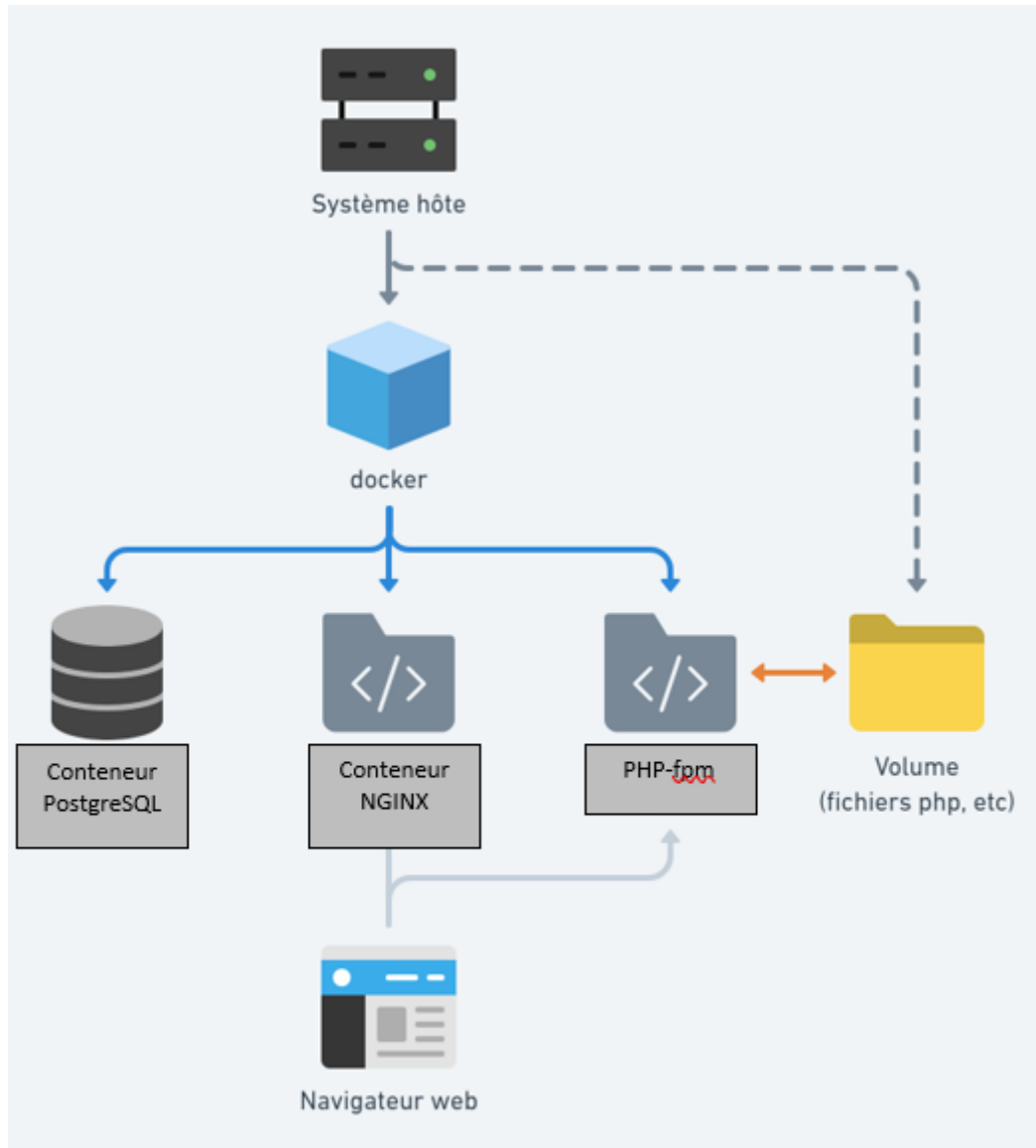
Les utilisateurs prévus sur le site sont :

- Des utilisateurs type, qui veulent participer à des challenges communautaire ou entre amis
- Les administrateurs et modérateurs du site
- Des écoles, qui veulent faire participer leurs élèves à des challenges
- Des entreprises, qui veulent au cours de certaines journées créer des challenges inter-entreprise.

II - L'architecture

L'architecture du site Défis tes copains est basée sur du micro-service grâce à l'utilisation de Docker. Avec celui ci le site tourne sous un serveur Nginx et avec une base de PostgreSQL

Le site est codé avec Symfony utilisant PHP 8.0.0 pour le Back et avec Materialize et Twig pour le front. Il y a aussi des fichiers JS et CSS.



III - Exigences du test

Pour la création du test de performance nous avons testé la connexion de plusieurs utilisateurs en même temps lorsque par exemple une école se connecte ou une entreprise .

Business Transactions	User Load	Response Time	Transactions per hour
Access Challenge Page	100	< 1.5 sec	150

IV - Environnement de test

Les tests vont être effectués sur le serveur de prod sous linux, possédant une mémoire RAM de 2 GO.

V - Planification des tests

Quel est le volume max d'utilisateur en temps réel ? Estimation à 100 Utilisateurs
Quel est le volume d'utilisateur pour les tests ? Tester sur 20/50/100 Utilisateurs

Type de test à prévoir pour le site Défis tes copains :

- Load Testing
- Spike Testing

Les metrics que l'on souhaite surveiller sont :

- Taux de requête en d'erreur
- Temps de réponse des requêtes

Les metrics qui définissent la réussite ou l'échec du site sont :

- Taux de requête en d'erreur
- Temps de réponse des requêtes

VI - Étape des tests

Étape des actions utilisateurs :

#STEP	Business Process Name : Création de défis (Challenges)
1	Page d'accueil
2	Connexion
3	Page de création de challenge
4	Page du challenge créé
5	Déconnexion
Jeux de donnée	Type de donnée : SQL Quantité de donnée : 100 insertions Provenance de la donnée : Fichier SQL provenant de la base
Préparation de la donnée	Création de la donnée via les fixtures Symfony, insertion dans la base de donnée les lignes de la table "challenge"

#STEP	Business Process Name : Inscription à un défis
1	Page d'accueil
2	Connexion
3	Chercher un challenge
4	S'inscrire à un challenge
5	Déconnexion
Jeux de donnée	Type de donnée : SQL Quantité de donnée : 100 insertions Provenance de la donnée : Fichier SQL provenant de la base
Préparation de la donnée	Création de la donnée via les fixtures Symfony, insertion dans la base de donnée les lignes de la table de jointure "challenge_user_register"

#STEP	Business Process Name : Consultation des défis
1	Inscription
2	Connexion
3	Page d'accueil
4	Page défis
Jeux de donnée	Type de donnée : CSV Quantité de donnée : 100 insertions Provenance de la donnée : Générateur de fichier csv avec des fausses données d'utilisateurs
Préparation de la donnée	Aucune

VII - Exécution des tests

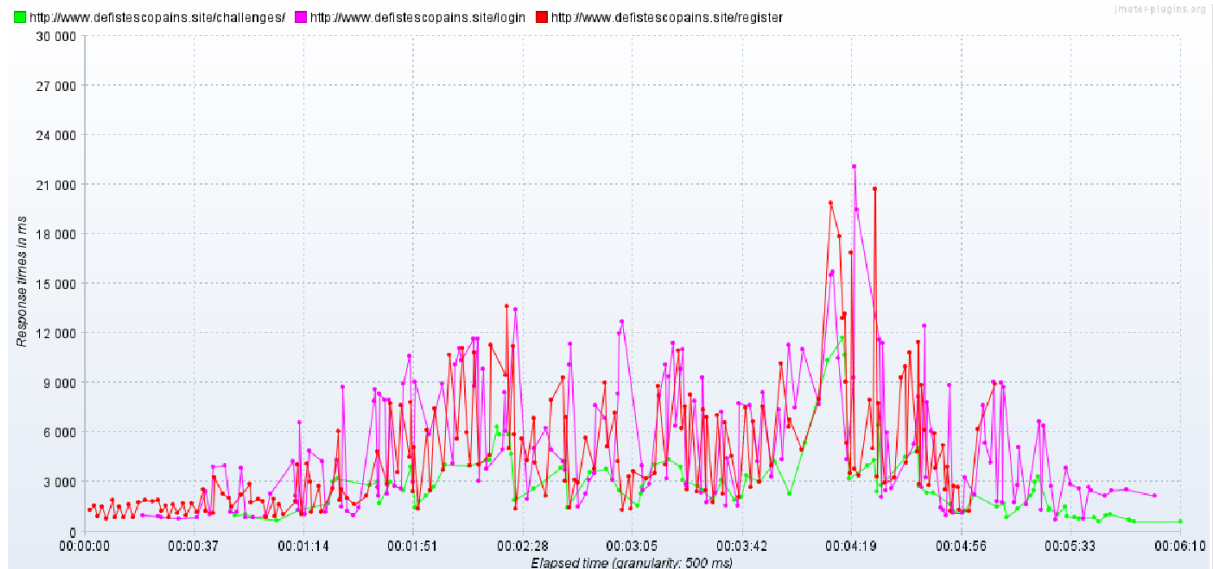
Scénario	Inscription + accéder à la page des défis
Cycle 1 - Run 1	Load Testing - Test 100 utilisateurs, monté en charge 600 secondes
Cycle 1 - Run 2	Spike Testing - Test 50 utilisateurs, monté en charge 1 seconde
Cycle 1 - Run 3	Spike Testing - Test 20 utilisateurs, monté en charge 1 seconde
Cycle 2 - Run 4	Load Testing - 50 utilisateurs, monté en charge 10 utilisateurs / 30 secondes
Cycle 2 - Run 5	Load Testing - 20 utilisateurs, monté en charge 10 utilisateurs / 30 secondes

	Détails des tests
Objectif	Vérifier le bon fonctionnement de l'accès à la page des défis, sur une arrivée rapide de 100 utilisateurs (run 1), 50 utilisateurs (run 2), et 20 utilisateurs (run 3).
Nombre de test	3
Durée	voir au dessus (ramp-up)
Script	1
Nom du scénario	Inscription + accéder à la page des défis
Volume d'utilisateur	100 vUser max
Critère d'entrée	Aucun
Validation des critères	<ul style="list-style-type: none">- Le temps de réponse doit être inférieur à 1.5 secondes- Le taux d'erreur doit être inférieur à 5%

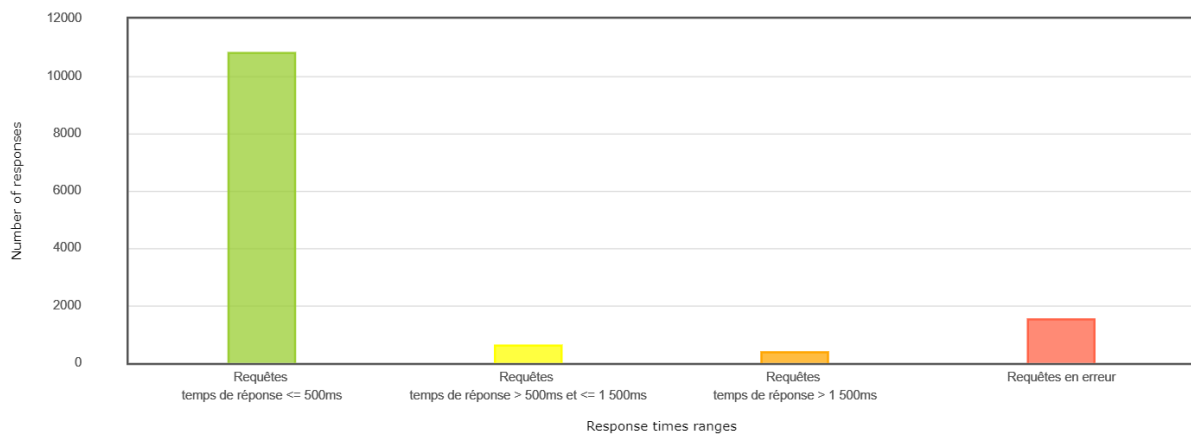
VIII - Résultats des tests

Inscription + accès à la page des défis

RUN 1 Résultat (load test 100 users - 600s ramp up)

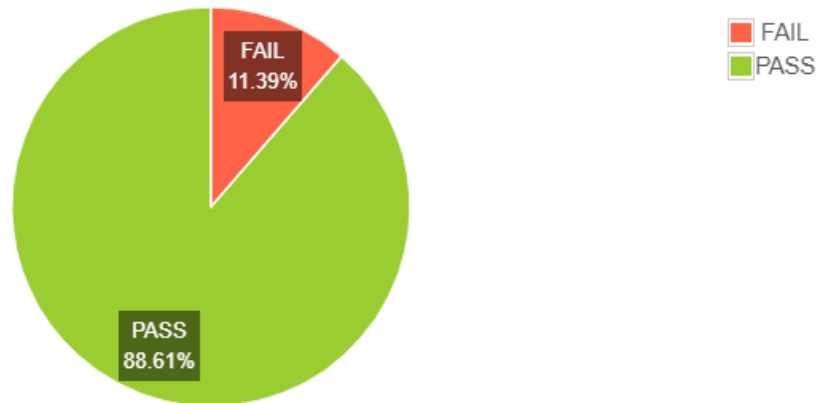


Response Time Overview

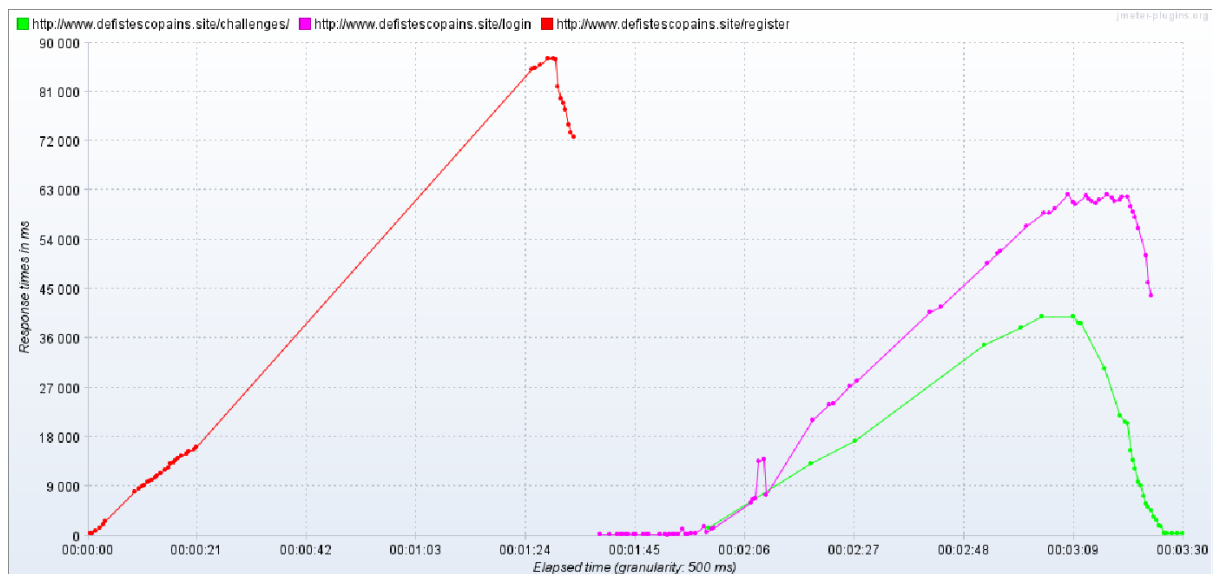


Requetes temps de réponse <= 500ms Requetes temps de réponse > 500ms et <= 1 500ms Requetes temps de réponse > 1 500ms Requetes en erreur

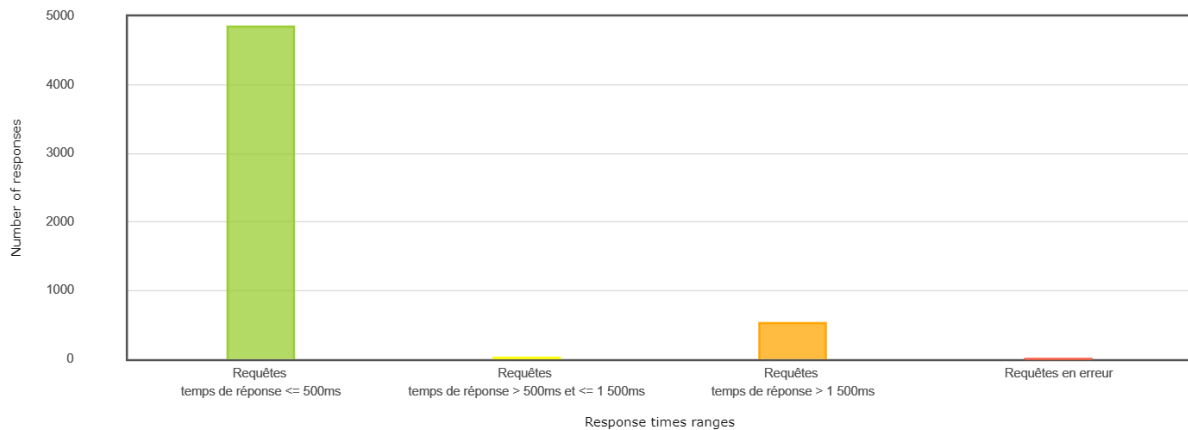
Requests Summary



RUN 2 Résultat (spike test 50 users - 1s ramp up)

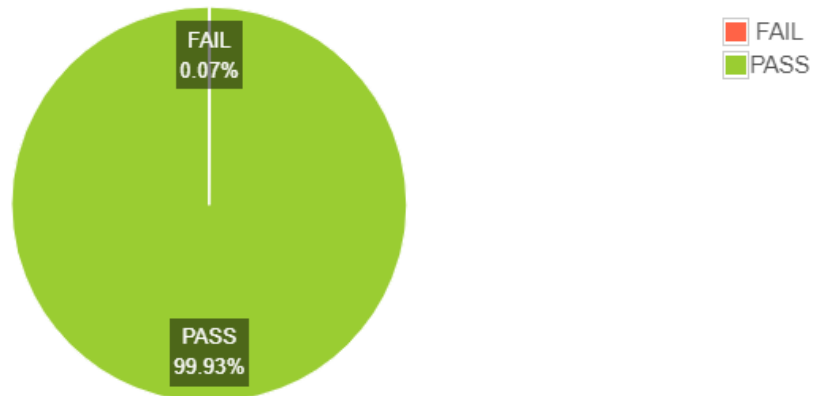


Response Time Overview

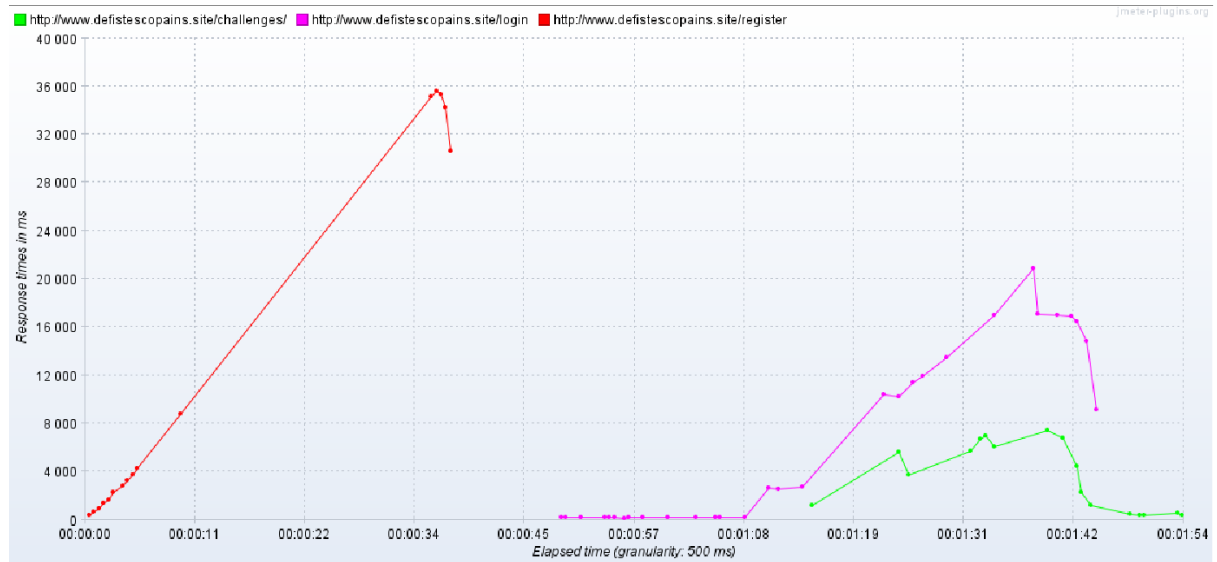


■ Requêtes temps de réponse <= 500ms ■ Requêtes temps de réponse > 1 500ms ■ Requêtes temps de réponse > 500ms et <= 1 500ms ■ Requêtes en erreur

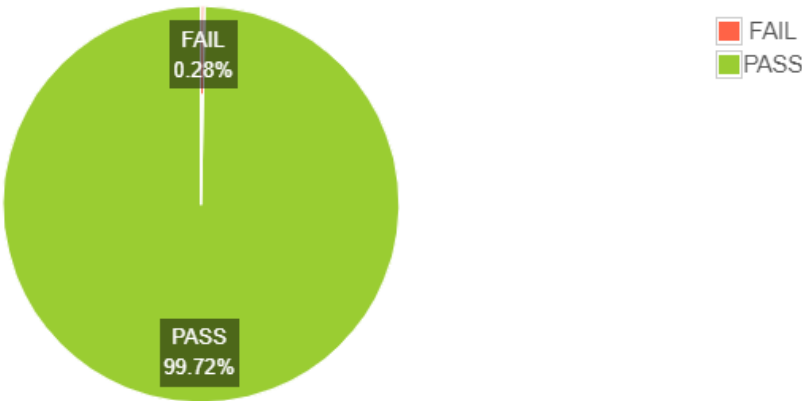
Requests Summary



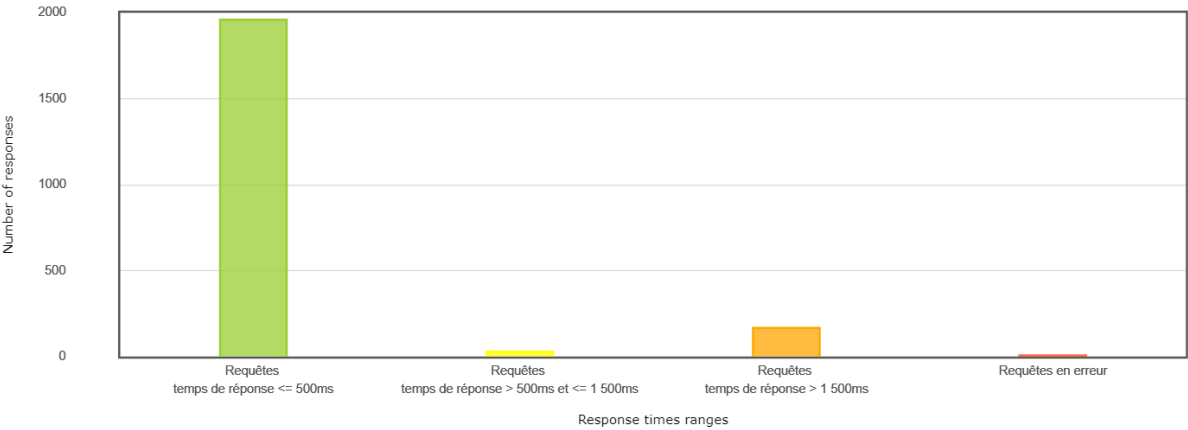
RUN 3 Résultat (spike test 20 users - 1s ramp up) :



Requests Summary

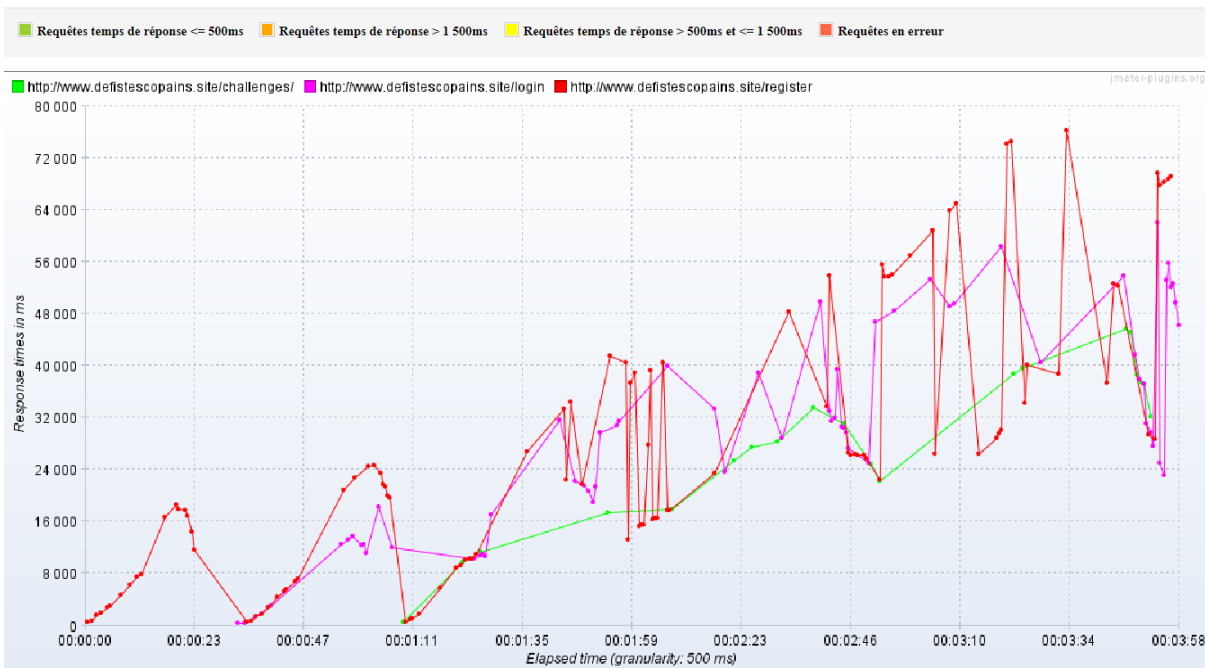
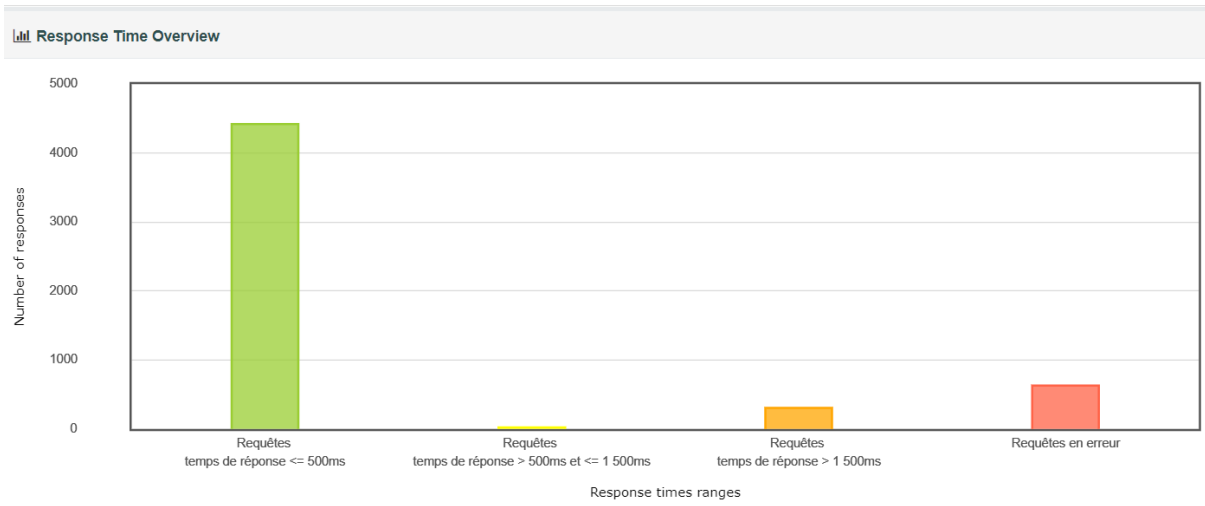
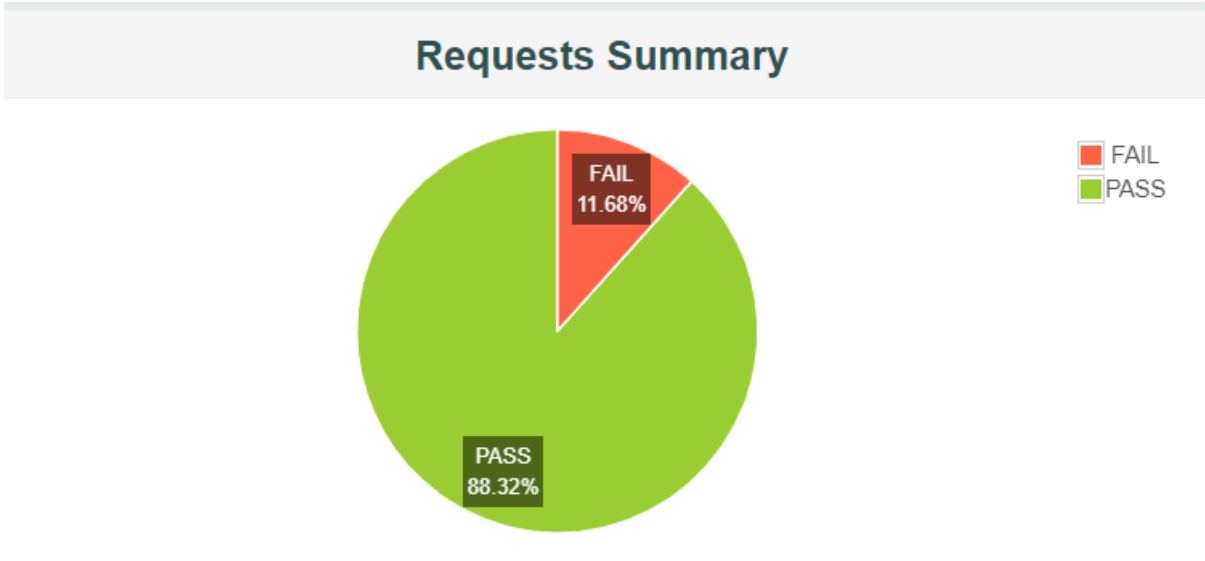


Response Time Overview

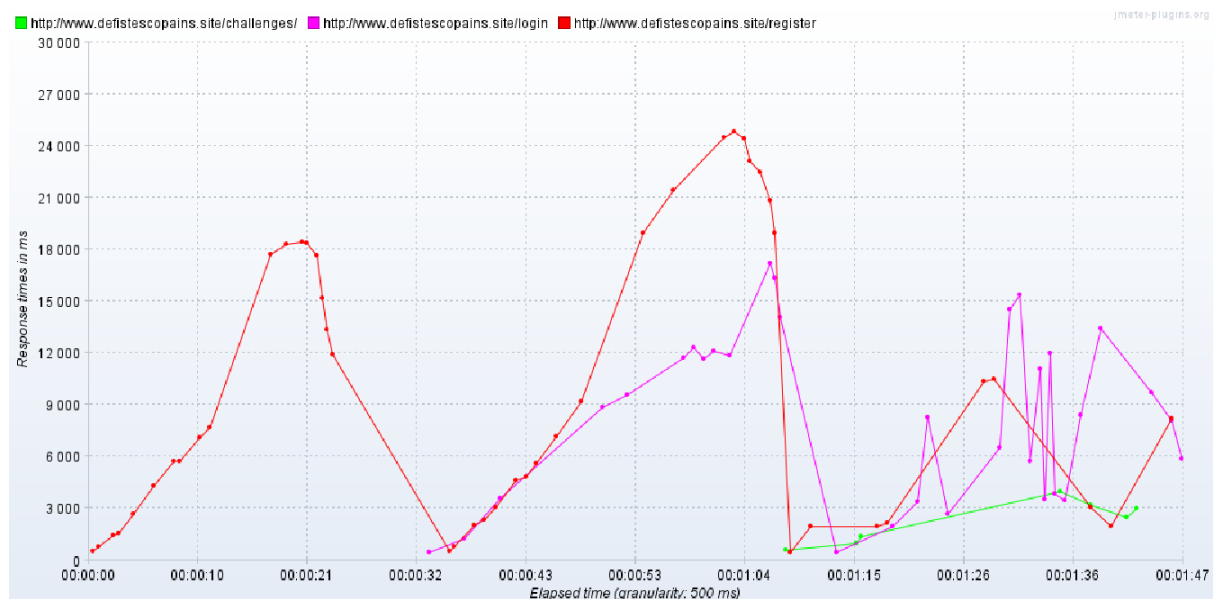
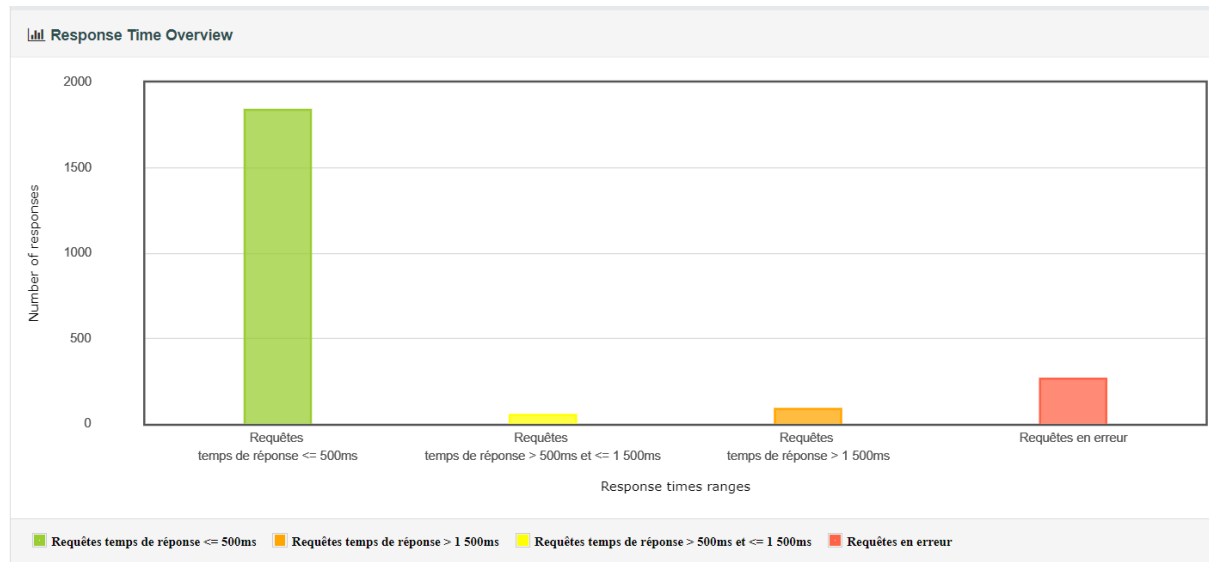
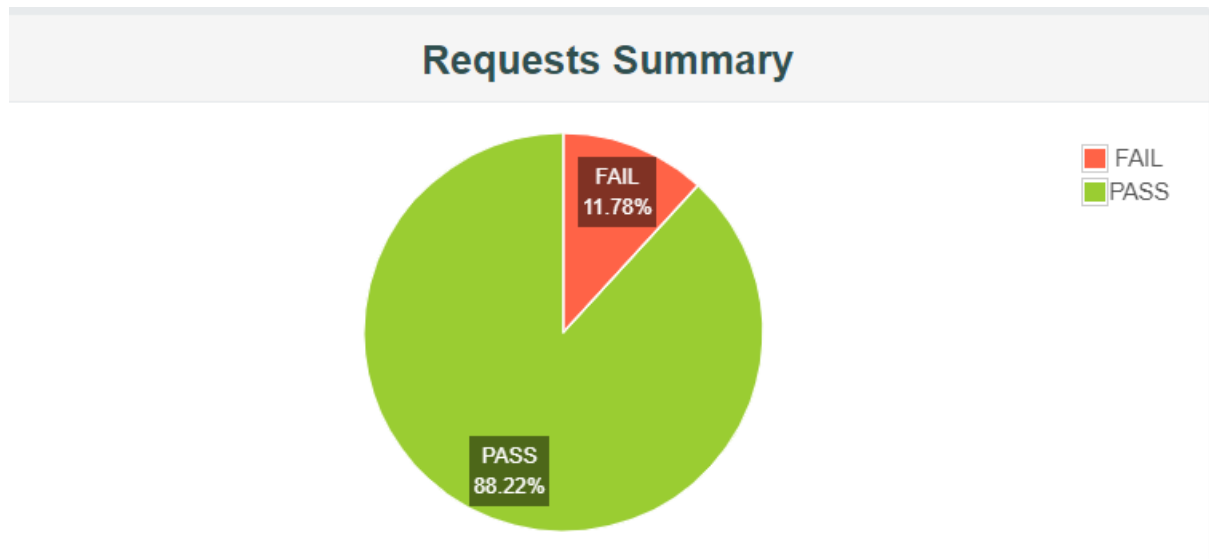


■ Requetes temps de réponse <= 500ms ■ Requetes temps de réponse > 1 500ms ■ Requetes temps de réponse > 500ms et <= 1 500ms ■ Requetes en erreur

RUN 4 - Load Testing - 50 utilisateurs, monté en charge 10 utilisateurs / 30 secondes :



RUN 5 - Load Testing - 20 utilisateurs, monté en charge 10 utilisateurs / 30 secondes :



VIII - Analyse et optimisation proposé

- L'infrastructure actuel n'est pas faite pour gérer 50-100 utilisateurs simultanément
- En cas d'augmentation du nombre d'utilisateurs, il faudrait prendre des machines + puissante
- Nous remarquons cependant que les tests sont rassurants: les temps de réponses sont prévisibles (nous laissent penser que le serveur est stable) et coïncident lors des spike testing par exemple où on remarque que les courbes de temps de réponses sont similaires qu'on envoie 20 ou 50 utilisateurs