DIAWADOH Locman

BOUKHATEM Mustapha

RARIVOMANANA Jédidia

**TESTS DE PERFORMANCE**

Nous avons choisi les 3 applications suivantes, du plus petit au plus grand en terme de trafic :

* Asbed.fr
* Footmercato.net
* Twitch.tv

**asebad.fr**

C’est un site vitrine sur un club de badminton. On y retrouve la présentation du club, sa localisation ainsi que son histoire. Nous pouvons aussi retrouver les règles du badminton et les tarifs pour rejoindre le club. Il y a aussi un formulaire de contact si on a des questions supplémentaires. L’objectif est de présenter le club aux personnes qui habitent dans les alentours et qui souhaitent faire du badminton. Les utilisateurs prévu sont de simple visiteurs.

**footmercato.net**

C’est un site internet entièrement dédié à l'actualité du foot et les rumeurs de transfert de joueurs. Il y a aussi une présentation détaillé sur chaque club avec l'effectif, le palmarès, les performances... C'est un site qui attend des utilisateurs passionné de foot, il n'y a pas de gestions de comptes... Dans le passés il y avait un forum mais il a été fermé. Il n’y a donc pas de parcours utilisateur.

**Twitch.tv**

Plateforme de streaming de contenu en temps réel ou à la demande, sous forme de clip ou de vidéo. Un utilisateur peut lancer un direct, en regarder un ou s’abonner à un autre utilisateur entre autre. Il y a un système de tchat avec interaction en temps réel entre les utilisateurs.

L’application a pour objectif de mettre en avant certaines chaînes, encourager à s’abonner car payant et également de constamment proposer du contenu même si un utilisateur met fin à son direct.

Les utilisateurs prévus sont les créateurs de contenu audiovisuel et toute personne en quête de divertissement, quel que soit son appareil.

Leur API web est en mesure de prendre en charge plus de 50 000 requêtes à la seconde en moyenne.

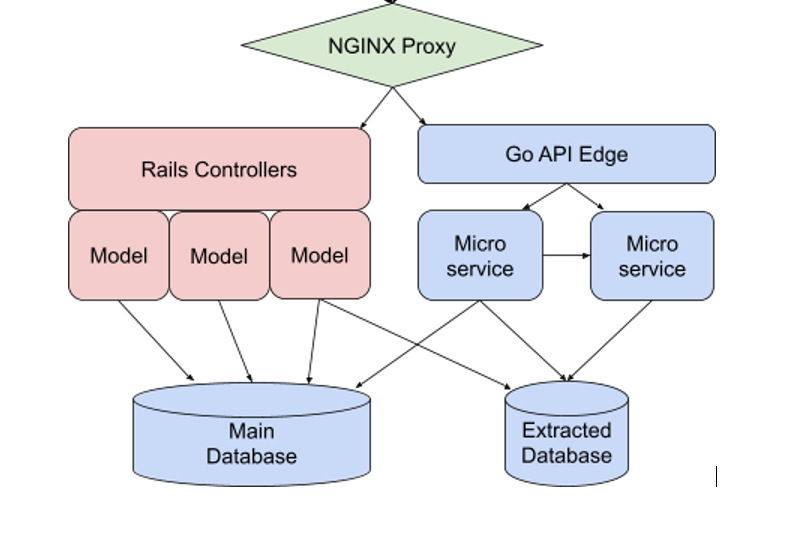
Le tchat délivre plus de 10 milliards de messages par jour.

**TECHNOS**

Ruby on Rails à l’origine, puis transition vers Go

+ TypeScript, Python, C++, Java/Kotlin et ObjectiveC/Swift sur d'autres plateformes

Jax pour gestion vidéos



**Les tests choisis**

Twitch connaît ses plus gros streamers, et rend sa plateforme capable de supporter un grand nombre de spectateurs, des milliers de messages sur le tchat ainsi que des donations régulières, le tout en simultané.

Nous allons partir sur un Spike Testing visant un petit streamer, que Twitch ne connaît pas encore, à l’aide d’un raid.

Un « raid » consiste à envoyer tous ses spectateurs sur un autre live.

Nous allons ainsi tester si un petit streamer est capable de subir un raid, et donc d’avoir un afflux de spectateurs en une très courte période.

Dans un deuxième temps, tous ces spectateurs doivent envoyer un message sur le tchat.

**Résultats attendus**

* Pas de perte de débit pour la vidéo en direct
* Temps de réponse de la plateforme équivalente après et avant raid
* Aucune latence sur le tchat suites aux messages

**Base**

Nous partons du principe que les utilisateurs ont une bonne connexion pour ne pas biaiser les tests.

**Les métriques**

Maximum active sessions – the maximum number of sessions that can be active at once