

StagingArea Onderzoek

Abstract

Dit verslag laat zien dat we een staging area moeten hebben en die gaan maken met MongoDB.

Inleiding

In dit onderzoek gaan we kijken hoe we het snelste en goed schaalbare staging area kunnen hebben. Het doel van de opdrachtgever is een bron zijn voor tussen-gebruikers die zelf een website willen maken. Dit is bij ons van belang omdat er verwacht wordt op 100.000 requests per minuut te zitten. Hiervoor hebben we dan ook de hoofdvraag "Wat voor database moeten we gebruiken als staging area?" bedacht. Om antwoord te kunnen krijgen hebben we ook een paar deelvragen bedacht namelijk:

- Welke eisen stelt de opdrachtgever?
- Welke niet functionele eisen stelt de opdrachtgever?
- Wat moet een Staging Area doen?
- Welke voordelen heeft een relationele database en een non-relatieve database?
- Welke non-relatieve databases zijn er?

Methode

Voor de deelvragen "welke eisen stelt de opdrachtgever?" en "Welke niet functionele eisen stelt de opdrachtgever?". Gaan we een klein interview houden met de opdrachtgever. Uit dit interview zouden de eisen duidelijk moeten worden. Hiervoor hebben we de vraag "Wat zijn uw eisen bij dit project?" gesteld aan de opdrachtgever. Als we na deze vraag nog steeds vragen hebben over de eisen zullen we die gaan stellen. Hier kunnen we nog geen vragen voor hebben omdat we niet weten wat er niet duidelijk zal zijn. Ook moeten deze vragen die we door stellen open vragen zijn.

De deelvraag "Wat zijn ontwerp principes die de database snelheid kunnen beïnvloeden?" Dit hebben wij onderzocht door het op het internet op te zoeken welke grote bedrijven dit al hebben toegepast en hoe zij dit hebben gedaan. Ook gaan we dit zelf opzetten om te testen. Door het zelf op te zetten kunnen we ook meteen kijken of het goed schaalbaar is.

Theoretisch kader

Wat moet een Staging Area doen?

De dingen die de staging area moeten doen:

- Moet JSON kunnen aanleveren binnen 2 seconden
- Moet goed schaalbaar zijn
- Het hoeft geen constraints te gebruiken
- Het hoeft geen procedures te gebruiken
- Het hoeft geen triggers te gebruiken

Door middel van deze criteria zouden we een snelle en schaalbare staging area krijgen. Het snel aanleveren van JSON is een van onze belangrijkste punt. Dit is omdat we van de opdracht gever te horen hebben gekregen dat er wel 100.000 requests per minuut aan zou moeten kunnen. Deze eisen van de opdracht gever kunt u vinden in het hoofdstuk *"Resultaten, Welke eisen stelt de opdracht gever"*.

De technische eisen die de staging area moet kunnen doen hebben we ook vast gesteld met wat voor kennis die we al hebben ontwikkeld in de ISE-courses.

Welke voordelen heeft een relationele database en een non-relationale database?

Om dit te kunnen beantwoorden hebben we voor de relationele database voorkennis gebruikt uit de ISE-courses. Hierdoor wisten we de voordelen van een relationele database al.

relationele database:

- Is goed in rekenen
- Kan constraints toevoegen
- Kan data goed bij elkaar halen

Voor deze non-relationale database punten hebben we onderzoek gedaan op het internet.

non-relationale database (Osman, 2023):

- hoge schaalbaarheid
- goed in het werken met grote hoeveelheden ongestructureerde of semi-gestructureerde gegevens

Welke non-relationale databases zijn er?

Je hebt 5 verschillende soorten noSQL databases.

1. Document Store Databases
2. Key-Value Stores
3. Wide-column Stores
4. Graph Databases
5. Time-Series Databases

Nu we de soorten op een rij hebben staan. Zien we dat 1 Document Store Databases voor ons van belang zijn. Hieronder vallen MongoDB, Couchbase als databases.

Resultaten

Welke eisen stelt de opdrachtgever?

Na een gesprek met de opdrachtgever hebben we de volgende eisen kunnen achterhalen:

- Moet in JSON of XML-format terug gegeven worden.
- Identificatie van gebruiker eindgebruiker of tussen-man.
- Een soort betaal systeem hebben(API key?).

Uit deze eisen kunnen we opmaken dat we een snel en schaalbaar systeem moeten hebben. Dit systeem moet ook een betaalsysteem hebben en een identificatie systeem. Dit betaalsysteem hoeven wij niet te realiseren, maar de uitbreiding daarvoor moet wel mogelijk zijn.

Welke niet functionele eisen stelt de opdrachtgever?

Na een gesprek met de opdrachtgever hebben we de volgende niet functionele eisen kunnen achterhalen:

- 100000 requests per minuut over de hele wereld.
- De eindgebruikers kunnen niet rechtstreeks tegen de database praten.

Discussie

Nu is er bij ons opgekomen uit het onderzoek dat we kunnen kiezen tussen 2 verschillende databases, MongoDB en Couchbase. Tussen deze 2 moeten we gaan kiezen. Met wat onderzoek online kwamen we er snel achter dat MongoDB meer functies heeft en goede ondersteuning van andere systemen. Hiervoor kiezen we MongoDB.

Conclusies

We gaan een non-relatieve database gebruiken omdat die aan alle eisen voldoen die we hebben gesteld voor de staging area. Ook hebben we gekozen om MongoDB te gebruiken hiervoor.

Referentielijst

- Yadav, S. (2023, 19 december). How to optimize your databases for best performance. Baseline.
<https://www.baselinemag.com/analytics-big-data/how-to-optimize-your-databases-for-best-performance/>
- Pmp, K. J. (2023, 15 oktober). Scaling Up Databases in Large Organizations: Strategies and Challenges. LinkedIn.
Geraadpleegd op 23 april 2024, van <https://www.linkedin.com/pulse/scaling-up-databases-large-organizations-strategies-challenges-joshi>
- Content Studio. (2024, 18 maart). Wat is database-sharding? Pure Storage.
<https://www.purestorage.com/nl/knowledge/what-is-database-sharding.html>
- Osman, J. (2023, 22 september). Relationale versus niet-relationale databases. AppMaster - Ultimate All-in No-code Platform. <https://appmaster.io/nl/blog/relationale-versus-niet-relationale-databases>