

Onderzoek naar monitoring mogelijkheden van NodeJS toepassingen in 2019.

Onderzoeksvoorstel Bachelorproef

Michiel Leunens¹

Samenvatting

Deze studie omvat een totaal onderzoek en uitwerking van de mogelijkheden om NodeJS toepassingen te monitoren. Dit enerzijds op gebied van bestaande tools en wat de mogelijkheden ervan zijn, alsook zelf een onderzoek doen naar wat er bekenen kan worden intern in het proces van NodeJS. De doelstelling van dit onderwerp zal de lezer helpen met het monitoren van NodeJS applicaties door te onderzoeken wat de best practices zijn.

Hier schrijf je de samenvatting van je voorstel, als een doorlopende tekst van één paragraaf. Wat hier zeker in moet vermeld worden: **Context** (Waarom is dit werk belangrijk?); **Nood** (Waarom moet dit onderzocht worden?); **Taak** (Wat ga je (ongeveer) doen?); **Object** (Wat staat in dit document geschreven?); **Resultaat** (Wat verwacht je van je onderzoek?); **Conclusie** (Wat verwacht je van van de conclusies?); **Perspectief** (Wat zegt de toekomst voor dit werk?).

Bij de sleutelwoorden geef je het onderzoeksdomein, samen met andere sleutelwoorden die je werk beschrijven. Vergeet ook niet je co-promotor op te geven.

Sleutelwoorden

Onderzoeksdomein. Webapplicatieontwikkeling — NodeJS — Monitoren

Co-promotor

Michiel Cuvelier² (KAYZR)

Contact: ¹ michiel.leunens.y7743@student.hogent.be; ² michiel@kayzr.com;

Inhoudsopgave

1	Introductie	1
2	State-of-the-art	1
3	Methodologie	1
4	Verwachte resultaten	2
5	Verwachte conclusies	2

1. Introductie

NodeJS is een Javascript framework wiens populariteit in de afgelopen jaar hard is toegenomen. Ontwikkelaars genieten van verschillende voordelen. Het werkt asynchroon, het is makkelijk schaalbaar en het is zeer portabel. Doordat het Javascript is, kan elk besturingssysteem gebruik maken van de krachtige backendfuncties van NodeJS. Dit maakt het een uitstekend framework voor webapplicaties te ontwikkelen. NodeJS is echter niet makkelijk om te monitoren doordat het asynchroon is opgebouwd. Het toepassen van de juiste technieken om te monitoren kan de slaagkansen van een project echter goed verhogen, net als de levenscyclus van de applicatie. Op welke manieren kunnen we het monitoren van zulke applicaties aanpakken? Welke software en tools worden hiervoor gebruikt? Welke technieken worden het best

toegepast? En kan er ook intern in het proces van NodeJS gekeken worden en deze informatie toegepast worden? En zijn al deze technieken drastisch veranderd sinds het framework werd uitgebracht in maart 2009?

2. State-of-the-art

Hier beschrijf je de *state-of-the-art* rondom je gekozen onderzoeksdomein. Dit kan bijvoorbeeld een literatuurstudie zijn. Je mag de titel van deze sectie ook aanpassen (literatuurstudie, stand van zaken, enz.). Zijn er al gelijkaardige onderzoeken gevoerd? Wat concluderen ze? Wat is het verschil met jouw onderzoek? Wat is de relevantie met jouw onderzoek?

Verwijs bij elke introductie van een term of bewering over het domein naar de vakliteratuur, bijvoorbeeld (Doll1954)! Denk zeker goed na welke werken je refereert en waarom.

3. Methodologie

Onderzoek doen naar de verschillende mogelijkheden van software en tools. Een steekproef maken van een aantal node-developers en via een enquête vragen met welke tools zij hun NodeJS applicatie monitoren, uitgebracht tussen 2009 en 2019. We kunnen erna kijken welke tools beter presteren

dan anderen dankzij het gebruik van de servers van Kayzr. We kunnen dit uitgebreider onderzoeken door:

- Monitoren van api calls volgens het aantal keren opgeroepen
- Monitoren van api calls volgens duratie tot een response gestuurd wordt
- Het gemak om errors op te slaan en later te debuggen/analyseren
- Monitoren van deze node process en hun taxatie op de server waar ze draaien (CPU, RAM, netwerk, ...)

Hier beschrijf je hoe je van plan bent het onderzoek te voeren. Welke onderzoekstechniek ga je toepassen om elk van je onderzoeksvragen te beantwoorden? Gebruik je hiervoor experimenten, vragenlijsten, simulaties? Je beschrijft ook al welke tools je denkt hiervoor te gebruiken of te ontwikkelen.

4. Verwachte resultaten

Er zal waarschijnlijk wel een tool uitspringen die op alle vlakken gemiddeld goed presteert. Dit zal wel de tool zijn die het meest populair is idk

Hier beschrijf je welke resultaten je verwacht. Als je metingen en simulaties uitvoert, kan je hier al mock-ups maken van de grafieken samen met de verwachte conclusies. Benoem zeker al je assen en de stukken van de grafiek die je gaat gebruiken. Dit zorgt ervoor dat je concreet weet hoe je je data gaat moeten structureren.

5. Verwachte conclusies

De wereld van Javascript is nog in volle groei, maar is toch al een pak maturder dan vroeger. We zien dat de tools beter zijn geworden. De concurrentie is enorm groot aangezien NodeJS een enorm populair platform is. Daardoor zal het noodzakelijk zijn om een gepaste monitoringstechniek toe te passen adhv software die we reeds kennen en vertrouwen.

Hier beschrijf je wat je verwacht uit je onderzoek, met de motivatie waarom. Het is **niet** erg indien uit je onderzoek andere resultaten en conclusies vloeien dan dat je hier beschrijft: het is dan juist interessant om te onderzoeken waarom jouw hypothesen niet overeenkomen met de resultaten.