CD Onderzoek

justin van de laar

2021

Inhoud

[Inleiding 2](#_Toc74991816)

[Woord Verwijzing 2](#_Toc74991817)

[Voorwoord 2](#_Toc74991818)

[Leeswijzer 2](#_Toc74991819)

[Wat is CD? 3](#_Toc74991820)

[Wat houdt CD in? 3](#_Toc74991821)

[Hoe pas ik CD toe in mijn eigen project? 3](#_Toc74991822)

[Voordelen en Nadelen van CD 3](#_Toc74991823)

[Verwijzingen 5](#_Toc74991824)

# Inleiding

## Woord Verwijzing

CD = Continuous Delivery   
CI = Continuous Integration

## Voorwoord

Dit document is een klein onderzoekje naar wat CD is. De main vraag voor dit onderzoek is wat is CD. Daarop zal ik ook nog kijken naar wat houdt CD in, Hoe pas je CD toe en als laatste ga ik ook kijken naar voordelen en nadelen. Het doel van het onderzoek is om een beter beeld te krijgen over CD inn het algemeen

## Leeswijzer

Geen speciale leeswijzer voor dit document

# Wat is CD?

## Wat houdt CD in?

CD staat in kort voor Continuous Delivery, hoewel CD wordt ook wel verward met Continuous Deployment. Alleen in dit geval is de proper benaming Continuous Delivery. CD is ook wel de stap die wordt uitgevoerd na CI.

CD is de stap die wordt uitgevoerd na CI en is meestal zo geregeld dat het automatische loopt. De meesten processen hebben het zo uitgeregeld dat als er een pipeline gebuild wordt en deze slaagt met testen en bouwen, dat de pipeline vervolgens wordt gepusht op de test/live server. Dit is handig omdat het natuurlijk tijd bespaart en niet meer handmatig wordt geüpload

Alleen wat houdt CD nou in kort eigenlijk direct in. CD gaat vaak gepaard met CI. CI Streeft erna dat developers vaak kleine stukjes code uploaden en nalaten checken door bijvoorbeeld een pipeline die automatische built. Nou stel je voor dat allemaal slaagt wat dan. Op het moment dat CI klaar is komt eigenlijk CD. CD wordt gebruikt om dan de gepusht code te brengen naar de omgeving waar het hoort hands down de naam *Continuous Delivery.*

CD helpt vooral ook developers die werken met meerdere omgevingen waar op wordt geproduceerd. Dus voor developers die niet alleen werken met een productie omgeving. CD is een geautomatiseerd proces en zorgt ervoor dus dat het naar de goede omgeving wordt gebracht

## Hoe pas ik CD toe in mijn eigen project?

Hoe pas je nou CD toe. Wel het begint eigenlijk vrij simpel. Als eerste zul je een omgeving nodig hebben waar de pipeline die uit de CI komt naar toe kan worden gebracht. Dat is eigenlijk stap 2 want voor stap 1 is het logische dat je een CI hebt opgezet.

Meeste voorbeelden tonen aan dat er in de pipeline die wordt gebruikt meestal een stukje code staat dat als de pipeline slaagt dat het automatische wordt gebracht naar de omgeving. Ook zijn er teams die er tools voor gebruiken om het CI/CD proces automatische laten verlopen.

Moest ik zelf een manier kiezen dan zou mijn voorkeur eerst uitgaan naar een tool die CD kan doen voor me. Wel zou ik dan willen kunnen zien wat de tool doet, dit met de reden zodat ik kan zien wat het doet en desnoods aan kan passen en kan leren wat er verandert. Zelf denk ik hier meer van te leren dan het gelijk te pakken met een stukje code die op internet staat.

## Voordelen en Nadelen van CD

Dan als laatst CD komt ook nog met een paar voordelen en nadelen om in rekening te houden. Ik zal beginnen met de voordelen:

* CD kan zorgen voor snellere software releases en development cyclus
* CD kan zorgen voor betere en sneller feedback voor debuggen
* Overal product kwaliteit en software kwaliteit
* Betere tijd naar de juiste software omgeving

Ook komt CD toch met een paar nadelen die misschien handig zijn om is over na te denken. Een paar nadelen zijn als volgt:

* Beperking in sommige taken. Taken als de medische sector en telecom moeten meer en heviger worden getest dan andere branches
* Verschillende omgeving. How wel dit ook een plus punt kan het ook snel leiden tot en min punt, dit komt vaak omdat er zo snel veel verschillende omgevingen zijn dat er toch een bug ongemerkt kan meelopen naar deployment.
* Tests. Ook hoe wel het meestal automatische kan het toch een paar problemen opleveren. Meestal kan een lak aan geautomatiseerde test ook leiden naar een lak aan vertrouwen waardoor developers er toch van afkijken CD te gebruiken. Daarbij kunnen sommige dingen gewoon niet geautomatiseerd getest worden en hebben dus soms echt een menselijke check nodig

Dit zijn maar een paar punten waar we makkelijke over heen kunnen kijken. In de lange duur voor grote projecten is het natuurlijk handig CI/CD te gebruiken. Alleen als je een klein hobby project heb dan kan ik er nog wel is anders over nadenken.

# Verwijzingen

*14 CD tools uptodate* . (2021). Opgehaald van Katalon: https://www.katalon.com/resources-center/blog/ci-cd-tools/

Sacolick, I. (2020, Janurai 17). *What is CI/CD?* . Opgehaald van infoworld: https://www.infoworld.com/article/3271126/what-is-cicd-continuous-integration-and-continuous-delivery-explained.html

*Wikipedia CD*. (sd). Opgehaald van wikipedia: https://en.wikipedia.org/wiki/Continuous\_delivery