

# Een vergelijkende studie tussen een modulaire, individuele workflow en een traditionele, teamsgebonden workflow bij een Agile framework binnen software development en de invloed hiervan op developers

Onderzoeksvoorstel Bachelorproef 2020-2021

Nick Lersberghe<sup>1</sup>

## Samenvatting

Agile is een framework dat in het laatste decennium aan een grote opmars bezig is. Meer en meer bedrijven stappen af van een inflexibel watervalsysteem ten voordele van het flexibele en dynamische Agile framework. 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. IT-projectmanagement — Agile — Software development

## Co-promotor

Jocelyn Geurts<sup>2</sup> (Chronos Group)

Contact: <sup>1</sup> nicklersberghe@gmail.com; <sup>2</sup> TBD;

## Inhoudsopgave

1	Introductie	1
2	Stand van zaken	1
3	Methodologie	2
4	Verwachte resultaten	2
5	Verwachte conclusies	2

## 1. Introductie

Agile is een framework dat in het laatste decennium aan een grote opmars bezig is. Meer en meer bedrijven stappen af van een inflexibel watervalsysteem ten voordele van het flexibele en dynamische Agile framework. Momenteel behoort het tot de standaarden van de industrie. Binnen software development elimineert het enkele van de meest courante problemen die verantwoordelijk zijn voor het falen van veel software projecten.

Het valt te bemerken dat er een grote variatie bestaat in hoe de principes van Agile exact worden toegepast. Frameworks zoals Scrum en Kanban omvatten dezelfde principes met een andere uitwerking. In dit werk zal er geprobeerd worden te achterhalen wat de voor- en nadelen van 2 dergelijke frameworks zijn en hoe deze gecombineerd kunnen worden teneinde een beter projectverloop.

## 2. Stand van zaken

De principes waar Agile op rust zijn relatief simpel. Deze werden in 2001 besproken en uitschreven door

een groep van 17 developers (Beck2001). De voor dit werk relevante principes zijn (geparafraseerd en vertaald):

- Omarm veranderende requirements
- Lever regelmatig werkende software
- Een constant tempo moet voor onbepaalde tijd vol te houden zijn
- De beste architectuur, requirements en designs komen van zelf-organiserende teams

”Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage”. (Beck2001). De klant is in staat om de processen van projecten te beheersen aan de hand van on-site interactie en requirements geven de huidige noden van de eindgebruikers waarheidsgetrouw weer (Kumar & Bhatia, 2012). Dit wordt gezien als het grootste voordeel van Agile ten opzichte van een klassiek watervalsysteem. Frequente communicatie met de opdrachtgever (‘De klant’) leidt tot een constante instroom van feedback. In combinatie met een ander voordeel (‘Lever regelmatig werkende software’, zie verder) is de analist beter in staat de wensen van de klant (of herzieningen hiervan) en de interpretatie van het ontwikkelingsteam (‘De developers’) gelijk te stellen (of hier zo dicht mogelijk bij te komen). Een bijkomend voordeel is hier dat er een constante herziening van het product is, wat zorgt dat foutopsporing sneller en gemakkelijker gebeurt (Imreh, Raisinghani e.a., 2011).

”Deliver working software frequently, from a couple of weeks to a couple of months, with a preference

to the shorter timescale”(Beck2001). Eén van de essenties van Agile werken, is het iteratief en incrementeel werken. In de praktijk vertaalt dit zich naar het werken in sprints.

### 3. Methodologie

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

Hier beschrijf je welke resultaten je verwacht. Als je metingen en simulaties uitvoert, kan je hier al mockups 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

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.

### Referenties

- Imreh, R., Raisinghani, M. e.a. (2011). Impact of agile software development on quality within information technology organizations. *Journal of Emerging Trends in Computing and Information Sciences*, 2(10), 460–475.
- Kumar, G. & Bhatia, P. K. (2012). Impact of agile methodology on software development process. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, 2(4), 46–50.