



UHASSELT

KNOWLEDGE IN ACTION

Faculteit Bedrijfseconomische Wetenschappen

master in de toegepaste economische wetenschappen: handelsingenieur in de beleidsinformatica

Masterthesis

Agile projectmanagement in hoger onderwijs: onderzoek en toepassing

Anthony Taravella

Scriptie ingediend tot het behalen van de graad van master in de toegepaste economische wetenschappen: handelsingenieur in de beleidsinformatica

PROMOTOR :

Prof. dr. Benoit DEPAIRE

COPROMOTOR :

Dr. Marijke SWENNEN



UHASSELT

KNOWLEDGE IN ACTION

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

2017
2018



Faculteit Bedrijfseconomische Wetenschappen

master in de toegepaste economische
wetenschappen: handelsingenieur in de
beleidsinformatica

Masterthesis

Agile projectmanagement in hoger onderwijs: onderzoek en toepassing

Anthony Taravella

Scriptie ingediend tot het behalen van de graad van master in de toegepaste economische wetenschappen:
handelsingenieur in de beleidsinformatica

PROMOTOR :

Prof. dr. Benoit DEPAIRE

COPROMOTOR :

Dr. Marijke SWENNEN

Woord vooraf

Deze masterproef vormt het sluitstuk van mijn opleiding tot Handelsingenieur in de Beleidsinformatica aan de Universiteit Hasselt. De reden van mijn keuze voor het onderwerp agile projectmanagement in het hoger onderwijs, is tot stand gekomen doordat dit onderwerp uit de lijst met voorgestelde onderwerpen mij het meeste aansprak. Aangezien agile zeer belangrijk is de bedrijfswereld en waarschijnlijk ook in mijn carrière, ben ik tevreden dat ik mij verdiept heb in dit onderwerp. De reden waarom ik gekozen heb voor een case study leek me leuk om te doen omdat het enerzijds mijn eerste keer was dat in aanraking kwam met onderzoek en anderzijds dit ideaal was voor het kaderen van mijn bevindingen.

Het opstellen van deze masterthesis was een zeer uitdagende en leerrijke ervaring, maar ik zou nooit tot dit resultaat gekomen zijn zonder de hulp van verschillende personen.

Allereerst wil ik mijn promotor prof. dr. Benoit Depaire en mijn co-promotor dr. Marijke Swennen bedanken. Dankzij de wekelijkse sturing van dr. Marijke Swennen ben ik tegen een constant tempo blijven werken en door haar feedback heb ik steeds de problemen die ik tegenkwam kunnen oplossen. De constructieve feedback van prof dr. Benoit Depaire heeft ervoor gezorgd dat ik nodige aanpassingen heb doorgevoerd waardoor mijn thesis één geheel is geworden.

Daarnaast wil ik mijn mama en mijn vriendin bedanken voor hun morele steun tijdens deze lange periode. Ik moet toegeven dat ik niet altijd de aangenaamste persoon was om bij te zijn wanneer deadlines van de deeltaken (van de thesis) naderden. Vervolgens wil ik de agile coach bedanken om me de kans te geven om een dag uit zijn werkleven mee te maken. Ten slotte wil ik alle andere personen bedanken die mij geholpen hebben op zowel moreel als inhoudelijk vlak.

Anthony Taravella

Juni, 2018

Samenvatting

Agile projectmanagement is een bekende projectmanagementmethodologie die al jaren gebruikt wordt door IT bedrijven. Aangezien er een opkomst is van deze methodologie in het hoger onderwijs, is het nodig om te onderzoeken hoe deze methodologie zo goed mogelijk kan worden toegepast in een hoger onderwijsomgeving. In deze masterthesis werd hier onderzoek naar gedaan door middel van een exploratieve case study. Ten eerste werd er door het raadplegen van literatuur onderzocht welke kenmerken deze methodologie heeft, welke traditionele projectmanagementmethodes er bestaan en werden enkele studies aangehaald waarbij een agile methode wordt toegepast in het hoger onderwijs. Hieruit kon worden afgeleid dat agile projectmanagement een introductie in het hoger onderwijs kent, maar dit voornamelijk wordt toegepast in opleidingsonderdelen rond softwareontwikkeling. Uit de literatuurstudie was ook af te leiden dat de agile methode Scrum toepasbaar is op projecten die niet software gerelateerd zijn. Het was net daarom interessant om te onderzoeken welke wijzigingen er moeten worden doorgevoerd aan projectwerk om agile projectmanagement zo goed mogelijk toepasbaar te maken in het hoger onderwijs.

Dit werd ten eerste gedaan door twee enquêtes af te nemen bij studenten aan de universiteit Hasselt bij het opleidingsonderdeel "agile business model design". De eerste enquête werd afgenomen aan het begin van dit opleidingsonderdeel en de tweede werd afgenomen in de zevende week van het tweede semester van het academiejaar 2017-2018. De resultaten omtrent het gebruik van de Scrum methode waren zeer positief bij de studenten. Verder werden er vragen gesteld om de voorkeur van de studenten voor bepaalde elementen van projectwerk te achterhalen. De vragen zijn opgebouwd op basis van bevindingen in de literatuurstudie en zijn gekoppeld aan de twaalf agile principes om te kijken of deze toepasbaar zijn op projectwerk in het hoger onderwijs. Het resultaat was dat al deze principes toepasbaar zijn in een hoger onderwijs context mits het doorvoeren van enkele wijzigingen zodat de focus niet ligt op het softwareontwikkelingsaspect. Dit toont aan dat agile projectmanagement toepasbaar is op projectwerk in het hoger onderwijs.

Ten tweede werd er een meeloopdag georganiseerd in een agile bedrijf om te weten te komen of een agile bedrijf zich volledig aan de agile principes houdt en om te achterhalen hoe de bedrijfscontext verschilt van de hoger onderwijscontext op vlak van projectwerk. Uit deze meeloopdag bleek dat het bedrijf zich niet 100% aan de agile principes houdt. Zo wordt er bijvoorbeeld niet dagelijks met de klant samengewerkt aangezien dit niet nodig blijkt. Ook kwam hier aan het licht dat het bedrijf niet één specifieke methode toepast maar eerder het agile denken in acht neemt. Verder was er ook een agile coach aanwezig op deze meeloopdag. Deze agile coach voerde voor bij enkele werknemers een simulatie uit om aan te tonen hoe de flow van het werk verbeterd kan worden. Ook had hij een persoonlijke inbreng voor deze masterproef. Zo legde hij de nadruk op het feit dat wie agile wil werken, zich niet moet vastklampen aan een specifieke methode maar de agile principes in het achterhoofd moet houden. Verder haalde hij aan dat er aan studenten zeker duidelijk gemaakt moet worden waarom en voor wie ze een bepaald project maken zodat de kans groter wordt dat de studenten intrinsiek gemotiveerd worden.

Tot slot werden de gegevens verzameld in de literatuurstudie, vergaard uit de enquêtes en uit de observatie, omgevormd tot richtlijnen die docenten (en studenten) kunnen volgen zodat agile

projectmanagement toepasbaar wordt in het hoger onderwijs. Hierdoor kan er ook een antwoord geformuleerd worden op de onderzoeksvraag, namelijk: "Op welke manier kan agile projectmanagement toegepast worden in het hoger onderwijs?". Agile projectmanagement is toepasbaar in het hoger onderwijs als de geformuleerde richtlijnen in acht worden genomen. De richtlijnen zijn terug te vinden in hoofdstuk vijf van deze masterproef. Ze tonen aan dat de docenten en studenten zich zullen moeten aanpassen om agile projectmanagement zo goed mogelijk toe te kunnen passen in het hoger onderwijs.

Inhoudstabel

WOORD VOORAF	I
SAMENVATTING	III
INHOUDSTABEL.....	V
LIJST VAN TABELLEN	VII
LIJST VAN FIGUREN.....	VII
HOOFDSTUK 1: INLEIDING	1
1.1 Onderzoeksdoelstelling	2
1.2 Onderzoeksaanpak.....	3
HOOFDSTUK 2: LITERATUURSTUDIE.....	9
2.1 Inleiding	9
2.2 Wat is een methodologie?	9
2.3 Traditionele projectmanagementmethodologie	10
2.3.1 De traditionele softwareontwikkelingsmethodologie	11
2.3.2 Beperkingen van de heavyweight/traditionele methodes.....	15
2.4 Agile softwareontwikkeling	16
2.4.1 Extreme Programming (XP)	18
2.4.1.1 Aspecten van XP	19
2.4.1.2 De onderliggende praktijken en filosofie van XP	19
2.4.2 SCRUM.....	21
2.4.2.1 SCRUM rollen	21
2.4.2.2 SCRUM artefacten	22
2.4.2.3 SCRUM meetings	23
2.5 Agile projectmanagement in het algemeen	24
2.5.1 Risico's verbonden aan APM met bijhorende reacties.....	26
2.5.1.1 Development process risks.....	26
2.5.1.2 Management process risks	28
2.5.1.3 Work environment risks	28
2.5.1.4 Customer risks	29
2.6 De algemene verschillen tussen traditionele en agile projectmanagement methodes.....	29
2.6.1 De verschillende principes waar naar moet gekeken worden bij het kiezen van een projectmanagementmethodologie	30
2.7 Agile projectmanagement in het hoger onderwijs	31
2.7.1 SCRUM in het hoger onderwijs	32
2.7.1.1 de studies kort toegelicht.....	33
2.7.1.2 De belangrijkste bevindingen van de studies	33
2.7.2 XP in het hoger onderwijs.....	36
2.7.3 Algemene conclusie studies	37
2.7.4 Agile Learning	37
2.8 Motivatie van studenten.....	38
2.8.1 Doelen	39
2.8.2 Waarde	39

2.8.3 Verwachtingen	40
2.8.4 Omgeving.....	40
2.8.5 De interactie tussen drie elementen.....	41
2.8.5 Strategieën om waarde en positieve verwachtingen te ontwikkelen.....	41
2.8 Conclusie literatuurstudie.....	42
HOOFDSTUK 3: AGILE PROJECTMANAGEMENT IN HET HOGER ONDERWIJS	45
3.1 Inleiding	45
3.2 Een onderzoek naar agile projectmanagement in het hoger onderwijs	45
3.2.1 Voorkennis van Agile projectmanagement door studenten	47
3.2.2 Ervaring met werken volgens Scrum binnen projectwerk	48
3.3 De principes van agile projectmanagement toegepast op het hoger onderwijs	53
3.4 Beperkingen van deze studie en verder onderzoek	72
3.5 Conclusie	72
HOOFDSTUK 4: AGILE PROJECTMANAGEMENT IN DE BEDRIJFSCONTEXT	75
4.1 Inleiding	75
4.2 Het agile bedrijf.....	75
4.3 De werking en elementen van het bedrijf vergeleken met het hoger onderwijs	75
4.3.1 Indeling van de werkvloer	76
4.3.2 Visueel management bord per departement.....	76
4.3.3 Het gebruik van agile methodes	77
4.3.4 Samenwerking met de klant	78
4.3.5 Werken aan een constant tempo	79
4.3.6 Communicatie voor dagelijkse coördinatie	79
4.3.7 Collaboratie en interactie.....	80
4.3.8 Motivatie	80
4.3.9 Overkoepelend 'Pacemaker' team	81
4.4 De inbreng van de agile coach	82
4.4.1 Okaloa flowlab simulatie (geleid door de agile coach).....	83
4.5 Beperkingen van deze studie en verder onderzoek	86
4.6 Conclusie	86
HOOFDSTUK 5: FORMULEREN VAN RICHTLIJNEN VOOR HET TOEPASSEN VAN AGILE PROJECTMANAGEMENT IN HET HOGER ONDERWIJS	89
HOOFDSTUK 6: CONCLUSIE.....	95
LITERATUURLIJST:	97
BIJLAGES	99

Lijst van tabellen

Tabel 1: De verschillen tussen traditionele en agile PM (Stoica et al., 2013) (Fernandez & Fernandez, 2008) ..	30
Tabel 2: Vragen die werden gesteld in de studie van Valle & O'Mara (2015).....	34
Tabel 3: De berekende p-waardes van de chi-kwadraattoets.....	50
Tabel 4: De richtlijnen voor het toepassen van agile projectmanagement in het hoger onderwijs	93

Lijst van figuren

Figuur 1: Het raamwerk van de exploratieve case study	6
Figuur 2: Volgorde van de watervalmethode (Marks, 2002)	12
Figuur 3: Deliverables van de watervalmethode (Marks, 2002).....	12
Figuur 4: Rational Unified Lifecycle (Awad, 2005)	14
Figuur 5: V-Model lifecycle (Balaji & Sundararajan Murugaiyan, 2012)	15
Figuur 6: Evolutie van de watervalmethode naar Extreme Programming (Beck, 1999)	18
Figuur 7: het Scrum framework (May et al., 2016)	22
Figuur 8: Voorbeeld: Burndown chart (Schwalbe, 2012)	23
Figuur 9: Impact van waarde en verwachting op leren en prestatie.....	39
Figuur 10: Interactief effect van de omgeving.	41
Figuur 11: Voorkennis ASD door studenten.....	47
Figuur 12: De ASD's waarvan de studenten al hebben gehoord	48
Figuur 13: De werking van het project	49
Figuur 14: Aantal aangeduide elementen/handelingen van Scrum.....	50
Figuur 15: De voorkeur voor de manier van werken aan een project.....	56
Figuur 16: De manier van werken gedurende het project.....	57
Figuur 17: De mate waarin meelifters een negatieve invloed hebben op een project	57
Figuur 18: De problemen die zich tijdens het project hebben voorgedaan	58
Figuur 19: De mate waarin een goede score behalen de student motiveert	59
Figuur 20: Een ranking van de factoren die studenten motiveren goed te presteren tijdens een project.....	60
Figuur 21: Een ranking van de methodes voor het samenstellen van groepen	61
Figuur 22: Een ranking van de methodes voor het aanpakken van meelifters	63
Figuur 23: Stellingen over het samenwerken in dezelfde ruimte	64
Figuur 24: Een ranking over het verkiezen van manieren van communicatie	65
Figuur 25: De werkelijke manier van communiceren tijdens het project.....	66
Figuur 26: De manier van werken bij één finale deadline	67
Figuur 27: Een ranking over de voorkeur omtrent het voorbereiden op een project.....	69
Figuur 28: De mate waarin een leider een negatieve invloed heeft op een project.....	70
Figuur 29: De mate waarin een leider een positieve invloed heeft op een project.....	70
Figuur 30: Een ranking over manieren om een proces knelpunt op te lossen	71
Figuur 31: Algemeen visueel management bord	77
Figuur 32: behandelen van onzekerheid (bron van de agile Coach)	83
Figuur 33: Het visualisatiebord van de Okaloa flowlab simulatie	84
Figuur 34: De arbeidsverdeling (bron van de agile coach).....	84

Hoofdstuk 1: Inleiding

Projecten zijn in de bedrijfscontext onmisbaar. Elk bedrijf doet projecten, en deze projecten worden in de meeste gevallen in een onzekere omgeving uitgevoerd waardoor er een grotere kans is op falen als deze niet goed worden aangepakt. Om dit risico te minimaliseren bestaat er projectmanagement. Projectmanagement wordt door het Project Management Instituut (PMI) gedefinieerd als: "De toepassing van processen, methodes, kennis, vaardigheden en ervaring om de projectdoelstellingen te bereiken die kunnen worden gedefinieerd in termen van output, resultaten of voordelen." (Banica, Radulescu, Rosca & Hagi, 2017).

Er zijn verschillende soorten projectmanagementmethodes die een team kan gebruiken om een project tot een goed einde te brengen. De meest gebruikte methodes zijn de traditionele projectmanagement methodes. Om een voorbeeld te geven is PMBOK (Project Management Body Of Knowledge) een zeer bekende traditionele methode. Zo bestaat er de PMBOK guide wat een gids is die ervoor zorgt dat er effectieve projectmanagement kan gedaan worden (Project Management Institute, 2017). Bij deze methode, net zoals bij de meeste andere traditionele projectmanagementmethodes, wordt de aanpak opgesplitst in vijf opeenvolgende fases van projectmanagement, namelijk: initiatie, planning, uitvoering, prestatie/monitoring en projectafsluiting (Banica et al., 2017). Dit betekent dat de meeste projecten uit één cyclus bestaan die deze vijf fases omvat.

Softwareontwikkelingsbedrijven maakten in het verleden ook steeds gebruik van zo een traditionele methode. De watervalmethode is hier de bekendste en is ook wel bekend als het lineair sequentieel levenscyclusmodel (Stoica, Mircea & Ghilig-Micu, 2013). Deze bestaat ook uit vijf fases, maar deze fases zijn gericht op het ontwikkelen van software. Deze fases zijn de volgende: definieer, design, programmeren (schrijven van code), testen en implementeren (Marks, 2002).

Het grote probleem dat zich hier voordoet is dat er bijna geen veranderingen mogelijk zijn gedurende het project. Het veranderen of herwerken van een project worden als de duurste aspecten van traditioneel projectmanagement beschouwd (Karlesky & Van Der Voord, 2008). Dit zorgt ervoor dat wanneer er veranderingen doorgevoerd worden, het prijskaartje fors zal toenemen. Aangezien een softwareproject vaak een langdurig project is, dat in een zeer veranderende omgeving wordt gedaan, is er een nieuw soort projectmanagementmethodologie ontstaan, namelijk de agile projectmanagementmethodologie. Agile projectmanagement wordt door Hass (2007) gedefinieerd als een hoog iteratief en incrementeel proces, waar de ontwikkelaars en project stakeholders actief samenwerken om het domein te begrijpen, identificeren wat er gebouwd moet worden en de prioriteit geven aan de functionaliteit. Dit betekent dat de klant een grotere rol speelt bij agile projectmanagement ten opzichte van de traditionele projectmanagementmethodologie, aangezien deze voortdurend wordt betrokken bij het proces en er dus beter kan omgegaan worden met veranderingen.

Er zijn verschillende methodes om aan agile projectmanagement te doen. De agile methodes worden door Melnik & Maurer (2005) gedefinieerd als: "mensgerichte hulpmiddelen en richtlijnen voor het bouwen van bruikbare software in onvoorspelbare, zeer volatiele omgevingen". Zo is Extreme

Programming (XP) een zeer bekende agile methode waarbij de ontwikkelingscyclus die bij de watervalmethode gebruikt wordt, opgesplitst wordt in veel kleinere ontwikkelingscycli die incrementeel op elkaar verder bouwen, zodat er makkelijk veranderingen doorgevoerd kunnen worden (Beck, 1999). Ook Scrum is een zeer bekende agile methode. Stoica et al. (2013) definieert Scrum als een iteratieve en incrementele methode waarvan het doel is om ontwikkelingsteams te helpen zich te concentreren op het behalen van doelen en het werk van minder belangrijke taken te minimaliseren. Deze methodes worden gestuurd door de twaalf leidende agile principes die in het agile manifesto vermeld staan, zodat ze flexibeler zijn en meer kans op succes hebben (Beck et al., 2001). Een meer gedetailleerde uitleg van de verschillende methodes is terug te vinden in de literatuurstudie.

In het onderwijs zijn projecten en groepswerken ook een veel voorkomend fenomeen waarmee iedere student al van jonge leeftijd mee in contact komt. In de meeste gevallen zijn dit projecten waarbij de einddeliverable is vastgelegd door het onderwijsteam, en de studenten dit zo nauw mogelijk moeten uitvoeren. Dit leunt sterk aan bij de traditionele watervalmethode waarbij het doel vooropgesteld is en het plan gevolgd wordt om dit doel te bereiken. Bij zo een project krijgen de studenten dus meestal uitleg over wat er gedaan moet worden en op welke manier, waardoor ze weten wat er van hen verwacht wordt. Verder krijgen ze dan een bepaalde tijd om dit project te voltooien. In het hoger onderwijs wordt er meer verwacht van de studenten. De complexiteit van het project en zelfstandigheid van een groep nemen hier toe. Zo kan het bijvoorbeeld zijn dat studenten in het hoger onderwijs iets moeten uitvinden en ontwikkelen waar de einddeliverable (per definitie) nog niet vastligt. Soms krijgen ze ook meer vrijheid bij de aanpak van het project, tenzij er een specifieke aanpak vermeld is.

In tegenstelling tot het feit dat agile projectmanagement voornamelijk bekend is in de bedrijfswereid, kent het sinds 2015 een opmars in het hoger onderwijs. Dit is af te leiden aan de hand van de jaren van uitgifte van de verschillende, maar relatief weinige, studies die er zijn gedaan omtrent agile projectmanagement in het hoger onderwijs. Echter zijn deze studies, waarin een agile methode gebruikt wordt, meestal gedaan in een opleidingsonderdeel voor softwareontwikkeling omdat deze methodes, zoals eerder vermeld, bedoeld zijn voor softwareontwikkeling. In de literatuur wordt er aangehaald dat een agile methode, zoals Scrum, ook kan toegepast worden in projecten waar geen softwareontwikkeling bij komt kijken. Dit toont aan dat agile projectmanagement ook gebruikt kan worden voor andere type projecten. Aangezien hier weinig onderzoek naar gedaan is, is het interessant om te weten te komen in welke mate het gebruik van agile projectmanagement toegelaten wordt in een hoger onderwijscontext en welke aanpassingen docenten en studenten zouden moeten doorvoeren om het toepasbaar te maken.

In deze thesis zal de focus niet liggen op één specifieke agile methode, maar zullen de twaalf principes uit het agile manifesto als basis dienen om na te gaan in welke mate agile projectmanagement toepasbaar is in het hoger onderwijs.

1.1 Onderzoeksdoelstelling

Deze thesis heeft als doel richtlijnen te formuleren die professoren (en studenten), scholen en/of onderzoekers kunnen gebruiken bij het toepassen of het verder onderzoeken van de agile

projectmanagementmethodologie in het hoger onderwijs. Deze richtlijnen zullen geformuleerd worden op basis van de bevindingen uit de literatuur en het onderzoek dat er gedaan werd.

In het tweede hoofdstuk, de literatuurstudie, zullen enkele bestaande traditionele en agile methodes en concepten uitgelegd worden, alsook de huidige toepassing van agile projectmanagement in het hoger onderwijs. In het derde hoofdstuk werd er onderzoek gedaan naar de voorkeuren van bepaalde elementen/handelingen van studenten tijdens een project. Dit werd gedaan aan de hand van enquêtes waarvan de vragen onderbouwd waren door de twaalf leidende agile principes die in de literatuurstudie aan bod komen. Op deze manier konden de twaalf principes vertaald worden naar de hoger onderwijscontext en kon er geconstateerd worden welke principes werkelijk toepasbaar zijn in het hoger onderwijs. Verder werd er in het vierde hoofdstuk onderzoek gedaan naar de verschillen tussen de hoger onderwijscontext en bedrijfscontext op vlak van projectwerk. Dit werd gedaan aan de hand van een observatie in een bedrijf en vragen die werden gesteld aan de medewerkers over de werking van dat bedrijf. Vervolgens zullen er richtlijnen geformuleerd worden die gebaseerd zijn op de bekomen resultaten uit hoofdstuk 2, 3 en 4. Deze richtlijnen zullen in een samenvattende tabel in hoofdstuk 5 geformuleerd worden. Tot slot bevindt de conclusie van deze masterthesis zich in hoofdstuk 6 waar er een antwoord op de centrale onderzoeksvraag geformuleerd zal worden.

De centrale onderzoeksvraag kan dus als volgt geformuleerd worden:

Op welke manier kan agile projectmanagement toegepast worden in het hoger onderwijs?

Er zullen ook drie deelvragen behandeld worden waarvan een antwoord op de eerste deelvraag geformuleerd zal worden in hoofdstuk 2, een antwoord op de tweede deelvraag in hoofdstuk 3 en een antwoord op de derde deelvraag in hoofdstuk 4. De drie deelvragen zijn de volgende:

- *Welke principes van agile zijn toepasbaar in het hoger onderwijs?*
- *Hoe verschilt de hoger onderwijscontext van de bedrijfscontext op vlak van projectwerk?*
- *Welke wijzigingen en aanbevelingen op vlak van projectwerk in het hoger onderwijs kunnen we doorvoeren om agile projectmanagement zo goed mogelijk toepasbaar maken?*

1.2 Onderzoeksaanpak

De onderzoeksmethode die in deze thesis gebruik zal worden is een exploratieve case study. Een case study wordt door Silverman (2005) gedefinieerd als volgt:

“Het basisidee is dat één geval (of misschien een klein aantal gevallen) in detail zal worden bestudeerd, gebruikmakend van welke methode dan ook passend lijkt. Hoewel er verschillende specifieke doeleinden en onderzoeksvragen kunnen zijn, is het algemene doel om een zo volledig mogelijk begrip van die zaak te ontwikkelen.”

De reden waarom het om een exploratieve case study gaat, is omdat er diepgaand onderzoek gedaan wordt om het basis theoretisch 'artefact' of model, dat verder beschreven wordt, aan te passen naar

een nieuw 'artefact' dat kan gebruikt worden om agile projectmanagement beter toepasbaar te maken in het hoger onderwijs.

De eerste databron die gebruikt zal worden als basis voor de case study is de literatuurstudie. Deze werd gedaan om bekend te geraken met de belangrijkste begrippen voor deze thesis zoals projectmanagement, projectmanagementmethodologieën, traditionele en agile projectmanagementmethodes. Deze literatuurstudie legt deze begrippen grondig uit zodat er geen verwarring kan ontstaan bij het gebruik van deze begrippen in het verdere verloop van de thesis.

Aan de hand van de literatuurstudie zal er worden aangehaald wat er al gepubliceerd is over deze begrippen en wat daarvan relevant is voor het onderzoek. Meer bepaald zal de literatuurstudie informatie bevatten over:

- Wat een methodologie is, wat projectmanagement is, welke traditionele en agile projectmanagement methodes er bestaan;
- Welke elementen (principes) de basis van agile projectmanagement bieden;
- Welke risico's er verbonden zijn aan het gebruik van agile projectmanagement met de bijhorende reacties;
- Waarin de traditionele en agile projectmanagement methodes van elkaar verschillen;
- Hoe agile projectmanagement zijn introductie in het hoger onderwijs kent;
- Wat het begrip 'agile learning' inhoudt, en;
- Wat de basis vormt van motivatie bij studenten en hoe motivatie vergroot kan worden.

Om deze informatie te vergaren werd gebruik gemaakt van wetenschappelijke artikels, masterthesissen, boeken en enkele internetbronnen. Hiervoor werd (hoofdzakelijk) gebruik gemaakt van platformen/databases zoals Google Scholar en EbscoHost. Verder werden er vier wetenschappelijke artikels voorzien door de co-promotor om bekend te geraken met agile projectmanagement. Deze artikels waren de basis voor het zoeken naar andere artikels, aangezien hier belangrijke kernwoorden, zoals Scrum en XP, in voorkwamen. Naast deze kernwoorden werd er ook gebruik gemaakt van combinaties van kernwoorden, zoals *agile* en *projectmanagement*, *Scrum* en *classroom*, *agile* en *higher education*, *agile* en *university*, *comparison of agile and traditional methodologies*, *agile* en *risks*. Tot slot werd er gekeken naar de referenties die werden gebruikt in de gevonden artikels. Hieruit zijn andere artikels alsook boeken gevonden die extra informatie opleverden. Er werd teruggedaan tot het jaar 1999 om een breed beeld van recente literatuur te bekomen. Verder werden uit deze lijst van bronnen alleen diegene gebruikt die een antwoord op de bovenstaande opsomming geven.

Op basis van de literatuurstudie zijn de twaalf algemene leidende principes van agile projectmanagement bekend. Deze worden gebruikt als het standaard theoretisch 'artefact' of model wat de basis vormt voor het verdere onderzoek. Ook werden er in de literatuurstudie enkele studies omtrent agile projectmanagement in het hoger onderwijs aangehaald. Aangezien het doel van deze thesis is om richtlijnen te formuleren over hoe we agile projectmanagement zo goed mogelijk kunnen toepassen in het onderwijs, zijn er uit de studies bepaalde elementen gehaald waarop bepaalde vragen uit de enquêtes (wat hierna volgt) zijn gebaseerd. Tot slot zijn enkele vragen van de enquêtes ook gebaseerd op de risico's die gepaard gaan met het implementeren van agile projectmanagement.

De tweede bron van informatie is bekomen door het uitvoeren van twee enquêtes bij studenten in het opleidingsonderdeel 'agile business model design' aan de Universiteit Hasselt. Wat hier direct kan worden opgemerkt is dat deze gekozen case geen random sample is, maar een theoretisch sample. Theoretische sampling wordt door Silverman (2005) beschreven als: "Het selecteren van groepen of categorieën om te bestuderen op basis van hun relevantie voor je onderzoeksvragen, je theoretische positie en vooral de uitleg of het account dat u ontwikkelt. Theoretische sampling houdt zich bezig met het construeren van een voorbeeld, wat theoretisch zinvol is, omdat het bepaalde kenmerken of criteria inbouwt die helpen bij het ontwikkelen en testen van je theorie en uitleg."

De respondenten stellen dus een theoretische sample voor aangezien de respondenten heel specifiek gekozen zijn omdat ze allememaal volgens de agile methode Scrum het project moeten uitvoeren en dus de twaalf leidende principes van agile in acht moeten nemen. Dit zorgt ervoor dat het doel is om de resultaten analytisch te veralgemenen in plaats van statistisch. Op deze manier wordt er gekeken of het basis artefact aanpassingen nodig heeft om specifiek in het hoger onderwijs toe te kunnen passen.

Zoals eerder vermeld zijn er uit de literatuurstudie, in de studies omtrent agile projectmanagement in het hoger onderwijs en de beschreven risico's en elementen verbonden aan APM, elementen gehaald waarop de vragen van de enquêtes die werden afgenomen bij studenten zijn gebaseerd. Deze vragen waren opgebouwd om te achterhalen wat de voorkeuren van studenten zijn tijdens een projectwerk, maar ook om te achterhalen of bepaalde ideeën veranderd waren na enkele weken te werken volgens de agile methode Scrum.

Er werd gebruik gemaakt van kwantitatieve en kwalitatieve vragen. De reden waarom er kwantitatieve vragen werden gesteld is omdat op deze manier bepaalde aspecten en elementen gevisualiseerd kunnen worden. Er werd ook gebruik gemaakt van kwalitatieve vragen om op deze manier meningen van de respondenten te kunnen achterhalen en de kwantitatieve aspecten toe te kunnen lichten. Zo werd er gebruik gemaakt van ja/nee vragen met toelichting, open vragen, meerkeuzevragen (met meestal een toelichting), selectievakjes (meerkeuzevragen waar meerdere antwoorden mogelijk zijn), vijfpuntige likert-schalen en rangschikking. Aan de hand van deze data konden de twaalf leidende principes vertaald worden naar het hoger onderwijs en kon er een antwoord gegeven worden op de eerste deelvraag: 'Welke principes van agile projectmanagement zijn toepasbaar in het hoger onderwijs op vlak van projectwerk?'

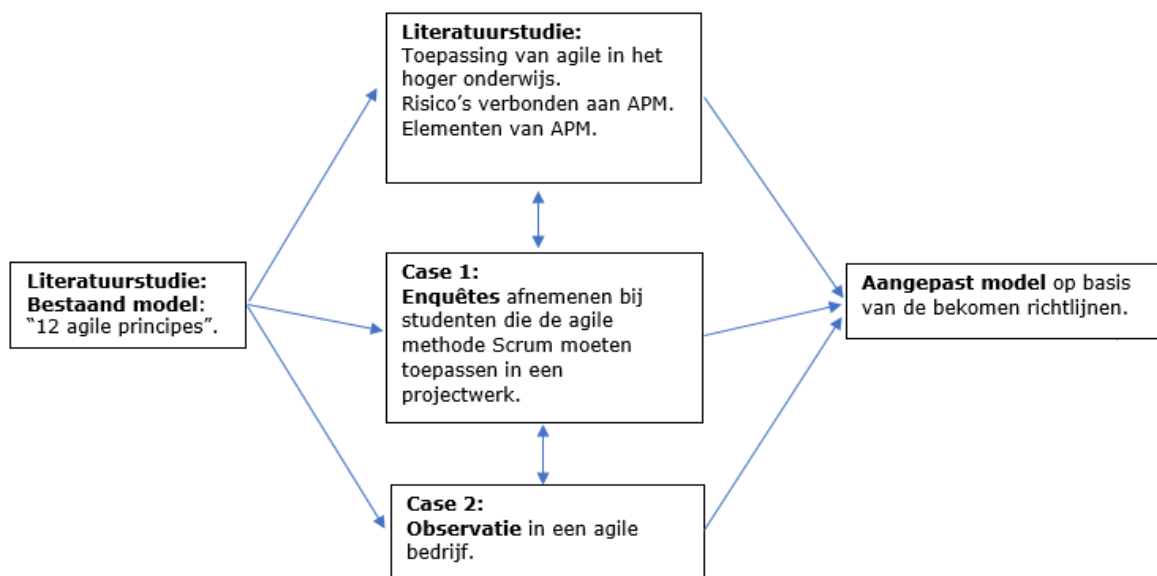
De resultaten die er uit de twee enquêtes gevonden zijn, zijn geanalyseerd met behulp van Microsoft Excel en het data-analyseprogramma R. Aan de hand van deze twee programma's zijn er grafieken gemaakt die ter visualisatie dienen doorheen hoofdstuk drie.

Als derde en laatste onderdeel van het onderzoek in deze thesis werd er gebruik gemaakt van een observatie. Zoals eerder vermeld, worden in de literatuurstudie de twaalf leidende principes van agile aangehaald alsook richtlijnen die bedrijven moeten volgen om op een agile manier te kunnen werken (grootte van teams, indeling van werkvloer, ...). Aangezien er van uit wordt gegaan dat deze principes ook werkelijk in een bedrijfscontext gevolgd worden, is er een meeloopdag georganiseerd in een bedrijf. Deze meeloopdag had als doel om te observeren of een bedrijf, dat zichzelf 'agile' noemt, ook volledig volgens deze principes werkt, maar ook om de verschillen op vlak van

projectwerk met het hoger onderwijs te onderzoeken. Ook bevat deze gekozen case geen random sample, maar een theoretische sample aangezien de werknemers werken volgens agile methodes en dus het basis artefact (de twaalf leidende principes) in acht houden.

Bij deze meeloopdag was er een professionele agile coach aanwezig die een simulatie deed samen met de werknemers. Het doel van deze simulatie was om de werknemers in te laten zien waar de problemen in de flow van het bedrijf zich kunnen voordoen en hoe deze kunnen worden opgelost. Verder is er geobserveerd hoe de werkvloer er uitzag en hoe de werknemers te werk gingen. Tot slot zijn er elementen van agile projectmanagement die niet zichtbaar waren door louter observatie. Hierdoor hebben enkele werknemers verteld op welke manieren het bedrijf werkt en heeft de agile coach bepaalde ervaringen gedeeld omtrent agiliteit. Op deze manier kan er een vergelijking gemaakt worden met de hoger onderwijscontext uit eigen ervaring en kan een antwoord op de tweede deelvraag geformuleerd worden, namelijk: "Hoe verschilt de hoger onderwijscontext van de bedrijfscontext op vlak van projectwerk?"

Deze observatie heeft dus als doel om na te gaan of de twaalf leidende principes van agile, ook werkelijk gevolgd wordt door bedrijven die zichzelf agile noemen en om te vergelijken hoe projectwerk in de bedrijfscontext verschilt van de onderwijscontext.



Figuur 1: Het raamwerk van de exploratieve case study

In figuur 1 is het raamwerk waarrond deze case study gebouwd is, weergegeven. Om samen te vatten wat hierboven vermeld is, kan er gekeken worden naar de figuur. Eerst is er de literatuurstudie die algemene begrippen uitlegt en het basis 'artefact', namelijk de twaalf leidende principes van agile beschrijft. Vervolgens is er een deel van de literatuurstudie waarin studies omtrent agile projectmanagement in het hoger onderwijs, de risico's verbonden aan APM en de elementen van APM worden beschreven en op basis waarvan bepaalde vragen van de enquêtes zijn geformuleerd. Verder is er de eerste case, het afnemen van de enquêtes bij de studenten, waaruit we voorkeuren van studenten bekomen omtrent projectwerk en de twaalf leidende principes kunnen vertalen naar een hoger onderwijscontext. Tot slot is er de tweede case, de observatie van het agile bedrijf waar

er gekeken wordt of de agile principes werkelijk worden toegepast en om de verschillen tussen de onderwijscontext te kunnen bekomen.

Door het samenvoegen van de bevindingen uit deze drie onderzoeksdelen, wat beter bekend staat als datatriangulatie, bevindt er zich in het laatste hoofdstuk een samenvattende tabel met richtlijnen die gebruikt kunnen worden om agile projectmanagement beter toepasbaar te maken in het hoger onderwijs. Deze richtlijnen zullen ondersteund worden door de twaalf leidende principes met de bijhorende vertaalde principes. Doordat de informatie uit drie verschillende bronnen komt, kunnen we deze bevindingen (analytisch) veralgemenen en kunnen deze gebruikt worden voor verder onderzoek.

Hoofdstuk 2: Literatuurstudie

2.1 Inleiding

Om een beter beeld te bekomen omtrent agile projectmanagement zal deze literatuurstudie bepaalde belangrijke begrippen, elementen en concepten uitleggen. Zo zullen er bekende traditionele projectmanagementmethodes en bekende agile projectmanagementmethodes uitgelegd worden. Hierna worden deze twee methodologieën met elkaar vergeleken. Verder zullen er studies omtrent de implementatie van agile projectmanagement in het hoger onderwijs aangehaald worden. Hieruit zullen belangrijke elementen gehaald worden die gebruikt werden in de twee onderzoeken die werden uitgevoerd in deze thesis. In sectie 2.2 zal er uitgelegd worden wat een methodologie is. Dit wordt gevolgd door het toelichten van bekende traditionele projectmanagementmethodes in sectie 2.3. Vervolgens zal in sectie 2.4 agile softwareontwikkeling uitgelegd worden en zullen de meest bekende agile projectmanagement methodes, namelijk Scrum en XP aangehaald worden. In sectie 2.5 zal agile projectmanagement in het algemeen uitgelegd worden en zullen enkele risico's die verbonden zijn aan het gebruik van agile projectmanagement (die van toepassing zijn op het hoger onderwijs) besproken worden. Verder zullen in sectie 2.6 de algemene verschillen tussen de traditionele en agile methodologieën voorgesteld worden. Hierna zullen in sectie 2.7 studies omtrent agile projectmanagementmethodes in het hoger onderwijs besproken worden. Deze studies betreffen de implementatie van Scrum en XP in het hoger onderwijs. Vervolgens zal er kort iets verteld worden over het concept van agile learning. Tot slot zal in sectie 2.8 het onderwerp motivatie bij studenten behandeld worden.

2.2 Wat is een methodologie?

Een methodologie is een set van richtlijnen of principes die kan worden toegepast op een specifieke situatie. Toegepast in een projectomgeving kunnen deze richtlijnen of principes verschillende zaken voorstellen. Zo kan het een lijst zijn met dingen om te doen, een specifieke aanpak, templates die gebruikt kunnen worden, formulieren en zelfs checklists die tijdens de levenscyclus van het project worden gebruikt (Rajamanickam, 2005).

Er zijn verschillende soorten methodologieën die in bedrijven worden toegepast. Zo bestaan er marketing & sales-methodologieën, wervingsmethodologieën, ontwikkelingsmethodologieën, operations & support-methodologieën en projectmanagementmethodologieën (Rajamanickam, 2005).

In deze thesis gaat de focus liggen op de verschillende projectmanagementmethodologieën. Projectmanagement wordt door het Project Management Instituut (PMI) gedefinieerd als: "De toepassing van processen, methodes, kennis, vaardigheden en ervaring om de projectdoelstellingen te bereiken die kunnen worden gedefinieerd in termen van output, resultaten of voordelen." (Banica, Radulescu, Rosca & Hagi, 2017).

In de volgende paragraaf zal er verder ingegaan worden op de meest voorkomende traditionele projectmanagementmethodologieën. Er zal verder steeds gesproken worden over methodes aangezien dit de aanpak voorstelt.

2.3 Traditionele projectmanagementmethodologie

Sinds de opkomst van het projectmanagementconcept in de jaren 1950, tot het heden, werden verschillende methodes ontwikkeld en toegepast. De meest voorkomende traditionele projectmanagementmethodes, zijn (Banica et al., 2017):

- PMBOK, en;
- PRINCE2.

PMBOK (Project Management Body of Knowledge) is ontwikkeld door het Project Management Instituut (PMI) en is gestructureerd rond vijf proces groepen en tien kennisgebieden. Deze vijf proces groepen zijn de opeenvolgende fases waaruit deze methode bestaat, namelijk: initiatie, planning, uitvoering, prestatie/monitoring en projectafsluiting (Banica et al., 2017; Fitsilis, 2007).

De planningsfase wordt als de belangrijkste fase beschouwd. Deze fase is gericht op het uitwerken van een rigoureuus plan van activiteiten, middelen, deadlines en levertijden, in overeenstemming met de voorwaarden voor het verzekeren van kwaliteit. Ook worden in deze fase de rollen en verantwoordelijkheden verdeeld in het projectteam. Ieder onderdeel dat voltooid is in de uitvoeringsfase, wordt in de prestatie/monitoringfase gecontroleerd op vlak van doelstellingen, kwaliteit en het beschikbaar budget. Verder worden de key performance indicatoren¹ periodiek geëvalueerd zodat de projectmanager afwijkingen van het ontworpen plan kan observeren en aanpassingen kan doorvoeren waar nodig. Tijdens de projectafsluiting wordt het product geleverd en zal de prestatie van het hele project geëvalueerd worden door de projectmanager alsook de klant (Banica et al., 2017).

Vervolgens zijn de tien kennisgebieden belangrijk om aan te halen aangezien deze, naast uit te leggen waarop PMBOK zich baseert, een algemeen beeld geven over wat projectmanagement is. Deze tien kennis/managementgebieden zijn de volgende (Fitsilis, 2007):

1. Integratie: Dit beschrijft de processen en activiteiten die verschillende aspecten van projectmanagement integreren.
2. Scope: Dit beschrijft de processen die verantwoordelijk zijn voor het controleren van de projectomvang.
3. Time: Dit beschrijft de processen aangaande het op tijd voltooien van het project.
4. Cost: Dit beschrijft de processen met betrekking tot de kosten van het project.
5. Kwaliteit: Dit beschrijft de processen die betrokken zijn bij het verzekeren dat het project zal voldoen aan de doelstellingen waarvoor het is uitgevoerd.
6. Human Resources: Dit bevat al de nodige processen voor het organiseren en managen van het project team.

¹ KPI (Key Performance Indicatoren) zijn een beperkt aantal indicatoren die aangeven hoe goed een bedrijf het doet in het algemeen en specifiek in het behalen van zijn doelen. <https://www.sixsigma.nl/woordenboek/kpi>

7. Communicatie: Dit beschrijft de processen met betrekking tot communicatiemechanismen van een project en heeft betrekking op de tijdige en juiste verzameling, verspreiding, opslag en uiteindelijke dispositie van projectinformatie.
8. Risk: Dit beschrijft de processen die betrekking hebben tot het project-gerelateerde risicomangement.
9. Procurement: Dit bevat al de processen die handelen met het verkrijgen van de producten en diensten die nodig zijn voor het voltooiën van het project.
10. Stakeholder: Dit is het laatst toegevoegde gebied in PMBOK en bevat de processen die nodig zijn om de stakeholders te informeren (wat bij Communicatie aan bod komt) alsook te betrekken bij het project.

Meer informatie over PMBOK kan worden terugvinden in de PMBOK guide². Deze gids vertelt hoe je effectief projectmanagement kan doen in elke industrie. In de laatste editie (6^{de} editie) zijn er ook elementen van hoe deze methode te implementeren in een agile (waarop later wordt teruggekomen) omgeving toegevoegd omdat hier door veel mensen vraag naar was (Project Management Institute, 2017).

PRINCE2 staat voor PProjects IN Controlled Environments en wordt beschreven als een gestructureerde methode voor effectief projectmanagement voor alle soorten projecten. Deze methode wordt erkend als een standaard die veel wordt gebruikt door de overheid in het Verenigd Koninkrijk en in de private sector. Deze wordt niet alleen gebruikt voor informatiesystemen of het ontwikkelen van software, maar de invloed van die industrie is wel merkbaar in deze methode. De versie uit 2002 heeft in het verleden een aantal grote veranderingen meegemaakt en is ontstaan uit de ervaring van tientallen projecten met de bijhorende projectmanagers en projectteams (Wideman, 2002).

Naast deze twee traditionele projectmanagementmethodes, zijn er nog andere traditionele/algemene projectmanagementmethodes, zoals Critical Path Method (CPM), Critical Chain Project Management (CCPM), Kanban, Lean, Six Sigma en Projects Integrating Sustainable Methods (PRiSM), maar deze zullen verder niet besproken aangezien deze buiten de omvang van deze thesis vallen.

2.3.1 De traditionele softwareontwikkelingsmethodologie

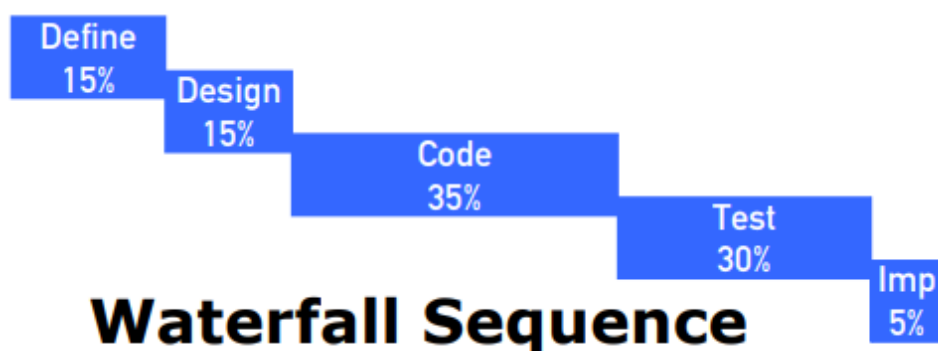
In deze thesis zal de focus op agile projectmanagement in het hoger onderwijs gelegd worden. Aangezien agile projectmanagement zijn oorsprong kent in softwareontwikkeling (hierop zal in sectie 2.4 teruggekomen worden), is het belangrijk om de voorlopers van de agile methodologie aan te halen zodat de verschillen tussen beide duidelijk worden. Deze traditionele softwareontwikkelingsmethodes, ook wel Heavy Weight Methodes (Awad, 2005) genoemd, zullen hierna beschreven worden.

De meest bekende en oudste methode is **de watervalmethode**. De watervalmethode is in het leven geroepen door Winston W. Royce in 1970. Deze methode is ook wel bekend als het lineair sequentieel levenscyclusmodel (Stoica, Mircea & Ghilig-Micu, 2013). Dit is een methode voor een

² <https://www.pmi.org/pmbok-guide-standards/foundational/pmbok/sixth-edition>

softwareontwikkelingsproces en benadrukt een gestructureerde progressie tussen gedefinieerde fases. Elke fase bestaat uit een voorafbepaalde set van activiteiten en een te behalen leverbaar doel dat voltooid moet worden voordat men naar een volgende fase kan gaan. De meest voorkomende vereenvoudigde structuur van de watervalmethode bestaat uit vijf fases zoals gevisualiseerd in figuur 2 (Awad, 2005).

In de eerste fase zullen de systeem- en softwarevereisten gedefinieerd worden en wordt bepaald wat het systeem zal doen, met andere woorden: het te verwezenlijke doel of product zal gedefinieerd worden. In de tweede fase wordt gekeken naar hoe het design van het systeem eruit zal zien. De derde fase bestaat uit het schrijven van code die gevolgd wordt door de testfase waarin het systeem getest wordt. De laatste fase is de implementatiefase waarbij er trainingen worden gegeven om het systeem te kunnen gebruiken en zal het systeem gedocumenteerd worden (Awad, 2005).



Figuur 2: Volgorde van de watervalmethode (Marks, 2002)

In projecten waarbij de watervalmethode wordt gebruikt, is het vaak zo dat 20-40% van de projecttijd wordt besteed aan de eerste twee fases, 30-40% van de projecttijd wordt besteed aan het eigenlijke programmeren en de overige tijd wordt gebruikt voor het testen en implementeren (Marks, 2002). Elk van deze fases heeft specifieke deliverables zoals reeds in de vorige alinea deels verteld is en zoals te zien is in figuur 3.

Waterfall Deliverables

Define	Design	Code	Test	Imp
Require ments.	Screens Database Objects Test Plan	UI Logic Reports	Test Scripts Defect Report User Feedback	Training Documentat ion
Project Management				
Project Charter, Status Reports, Change Requests				

Figuur 3: Deliverables van de watervalmethode (Marks, 2002)

Volgens Banica et al. (2017) wordt de watervalmethode gekenmerkt door acht opeenvolgende fases (conceptie, initiatie, analyse, ontwerp, opbouw, testen, implementatie en onderhoud), elk met rigoureuze planningsactiviteiten, duidelijk gedefinieerde doelstellingen en duidelijk gestructureerde tijdlijnen. Deze fases zijn iets verfijnder ten opzichte van de fases vermeld door Awad (2005) maar komen in het algemeen overeen met elkaar.

Deze methode heeft twee vaak voorkomende oorzaken van projectmislukking: Enerzijds laat het geen dynamische benadering toe, wat wil zeggen dat het moeilijk is om bepaalde zaken te wijzigen gedurende het proces. Anderzijds wordt het testen van het volledige systeem pas uitgevoerd na het voltooien van alle fasen. Dit betekent dat als er tijdens het proces iets foutloopt, dit pas zal opgemerkt worden in de testfase (Banica et al., 2017).

De auteur Awad (2005) meent dat op het vlak van softwareontwikkelingsmethodes de watervalmethode de bekendste is, maar naast deze zijn er volgens hem nog twee andere traditionele softwareontwikkelingsmethodes die significant zijn en veel worden gebruikt. Deze twee methodes zijn de Rational Unified Process methode en de Spiral Model methode. RUP zal vervolgens uitgelegd worden aangezien dit een belangrijke methode is waaruit de agile methode Extreme Programming veel inspiratie heeft gehaald (hier wordt op teruggekomen bij XP). Uitleg over het Spiral Model biedt geen meerwaarde voor deze thesis en zal dus niet worden uitgelegd. Meer informatie omtrent het Spiral Model is terug te vinden in Boehm (1988).

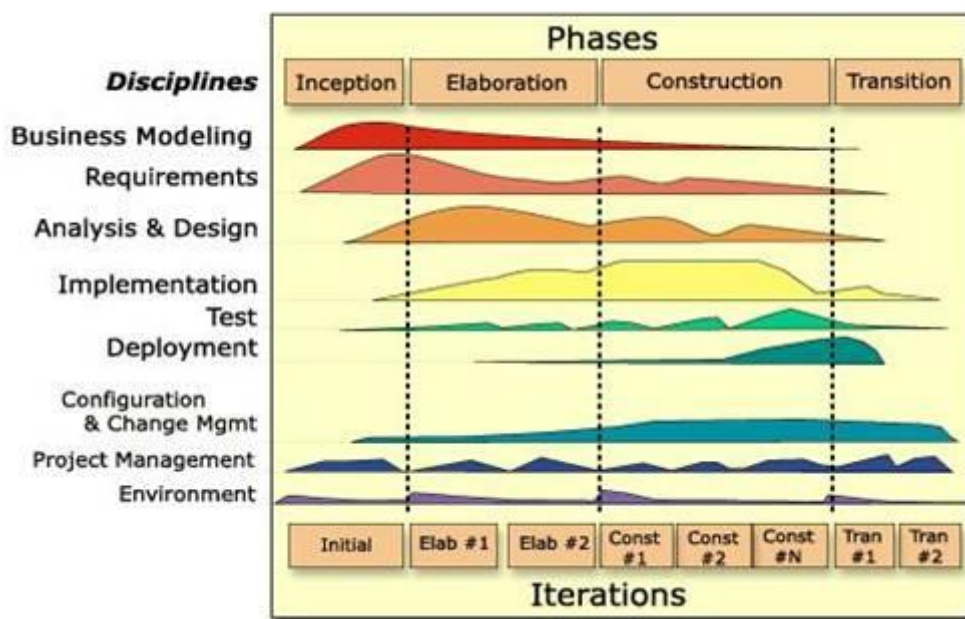
Rational Unified Proces (RUP) wordt door Krol & Kruchten (2003) opgesplitst in drie zeer verschillende zaken. Ten eerste is RUP een aanpak voor het ontwikkelen van software, die iteratief, architectuurgericht en use-case-driven is. Het gebruikt een iteratieve aanpak, wat wil zeggen dat het werkt met sequenties van incrementele stappen. Zo zal elke iteratie sommige, of de meeste elementen van ontwikkeling bevatten (vereistenanalyse, design, implementatie, testen, ...). Vervolgens maakt het gebruik van een op componenten gebaseerde architectuur dat een systeem creëert dat makkelijk uit te breiden is, dat software hergebruik bevordert en wat intuïtief verstaanbaar is. Verder maakt het gebruik van use-cases, wat een lijst met acties of event stappen voorstelt die de interacties tussen een rol en een systeem definiëren. Het voorziet ook verschillende richtlijnen en templates voor verschillende documentatie. Over het softwareontwikkelingsaspect is er veel terug te vinden in whitepapers en boeken, maar de meest volledige set van informatie is terug te vinden in het RUP-product zelf.

Ten tweede is RUP een goed gedefinieerd en goed gestructureerd software-engineeringproces. Het definieert duidelijk projectmijlpalen, wie verantwoordelijk is voor wat, hoe dingen worden gedaan en wanneer deze dingen gedaan moeten worden. Het maakt gebruik van zichtbaar modelleringssoftware zoals UML (Unified Modeling Language), wat code voorstelt als een schematische notatie zodat mensen zonder of minder technische ervaring toch een goede input kunnen bieden.

Tot slot is RUP ook een procesproduct dat een aanpasbaar procesraamwerk voor software-engineering biedt. Zo kan het RUP-product geconfigureerd worden om kleine of grote teams en gedisciplineerde of minder formele benaderingen van ontwikkeling te ondersteunen. Het laat ook toe om eigen 'best practices' toe te voegen en ervaringen en artefacten te delen met peers en experts (Krol & Kruchten, 2003).

Om terug te komen op het software-engineeringproces, werkt RUP met een project lifecycle die bestaat uit vier fases die elk een mijlpaal en een goed gedefinieerde set van objectieven bevatten. Deze vier fases worden weergegeven op figuur 4 en hieronder uitgelegd (Awad, 2005):

- Inception: Aan het einde van dit proces moet er een business case gemaakt zijn, de haalbaarheid van het project beoordeeld zijn en moet de reikwijdte (scope) van het ontwerp vastgesteld worden.
- Elaboration: In deze fase zou een basisarchitectuur moeten zijn geproduceerd en een bouwplan zijn overeengekomen. Bovendien vindt er een risicoanalyse plaats en moeten de risico's die als groot worden beschouwd, zijn aangepakt.
- Construction: Dit proces produceert een beta-release-systeem. Een werkend systeem zou beschikbaar moeten zijn en het moet voldoende zijn voor voorlopige testen onder realistische omstandigheden.
- Transition: Het systeem wordt geïntroduceerd bij de belanghebbenden en beoogde gebruikers. Het wordt geaccepteerd wanneer het projectteam en de belanghebbenden het erover eens zijn dat de doelstellingen die in de aanvangsfase (inception) zijn overeengekomen zijn bereikt en dat de gebruiker tevreden is.

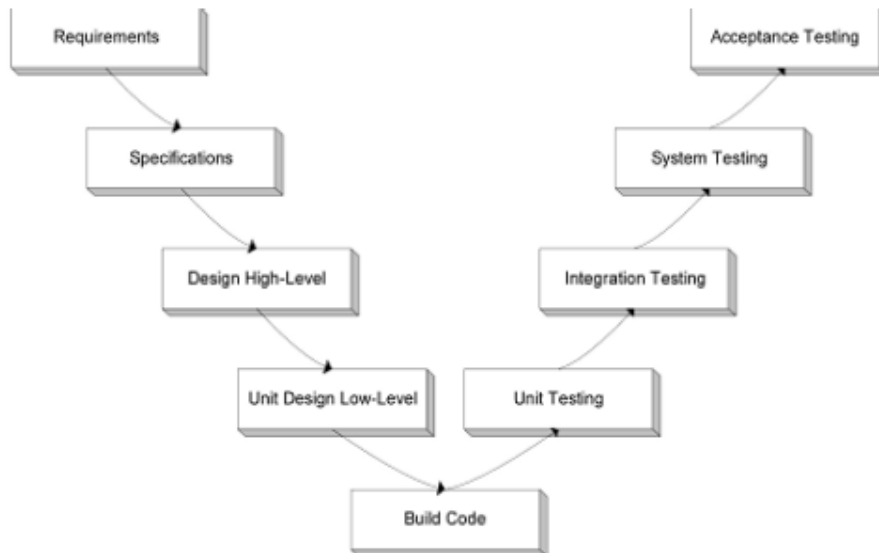


Figuur 4: Rational Unified Lifecycle (Awad, 2005)

Tot slot is er te zien op de figuur dat de iteraties binnen de bepaalde fases blijven waardoor er per iteratie nadruk gelegd wordt op de activiteiten binnen de bepaalde fase. Zo zal het moeilijk zijn om aanpassingen te doen, die nog zouden moeten doorgevoerd worden in fase 1, wanneer het proces zich in een latere fase bevindt.

In de literatuur wordt er ook regelmatig gesproken over het V-model (Validation & Verification model) wat ook een bekende traditionele softwareontwikkelingsmethode voorstelt. Deze methode is een uitbreiding van de watervalmethode. Ten opzichte van de watervalmethode werd deze niet op een lineaire as ontworpen, maar zullen de fases terug naar boven gaan zodra de coderingsfase gebeurd

is. Om dit te verduidelijken kan er gekeken worden naar figuur 5. Op deze figuur is te zien dat wanneer de code geschreven is, er voor elk deel (aan de linkerzijde) een bijhorende test gedaan wordt. Op deze manier is er een relatie tussen elke ontwikkelings- en testfase. Meer informatie over het V-model is terug te vinden in (Balaji & Sundararajan Murugaiyan, 2012).



Figuur 5: V-Model lifecycle (Balaji & Sundararajan Murugaiyan, 2012)

2.3.2 Beperkingen van de heavyweight/traditionele methodes

De grootste beperking van de heavyweight/traditionele methodes is het feit dat er bijna geen veranderingen mogelijk zijn gedurende het project. De vaardigheid om te reageren op veranderingen bepaalt vaak het succes of de mislukking van een softwareproject (Awad, 2005). Het veranderen of herwerken van een project worden als de duurste aspecten van traditioneel projectmanagement beschouwd. Er wordt geprobeerd om verandering tegen te gaan door planning, design en documentatie van een project op voorhand uitgebreid uit te voeren. Als er toch nood is aan verandering heeft dit volgens conventionele projectmanagementwijsheid te maken met het onvoldoende uitwerken van de planning, het design en de documentatie (Karlesky & Van Der Voord, 2008).

Het behandelen van complexe projecten is nog een limitatie van de heavyweight methodes. De traditionele aanpak waarbij alles eerst wordt gepland en vervolgens dit plan gevolgd wordt, werkt goed voor stabiele en minder complexe situaties, maar voor moeilijke projecten zal deze techniek niet gepast zijn. Dit omwille van het feit dat er bij complexe situaties vaak veranderingen aan het product moeten gebeuren waardoor er afgeweken wordt van het vooropgestelde plan (Awad, 2005).

2.4 Agile softwareontwikkeling

In tegenstelling tot de heavyweight of traditionele projectmanagementmethodes is er een nieuw soort van methodes in het leven geroepen: de agile-methodes, ook wel lightweight methodes genoemd. De agile methodologie kent zijn oorsprong in de wereld van softwareontwikkeling en presenteren nieuwe, niet-traditionele manieren om complexe producten en systemen te bouwen (Rajamanickam, 2005). Volgens Melnik & Maurer (2005) worden agile methodes gedefinieerd als: "mensgerichte hulpmiddelen en richtlijnen voor het bouwen van bruikbare software in onvoorspelbare, zeer volatiele omgevingen."

Agile methodes zijn in populariteit gestegen sinds de publicatie van het *agile software development manifesto*. Van 11 tot 13 februari 2001 zijn zeventien procesmethodologen samengekomen. Deze mensen presenteerden verschillende lightweight methodes zoals SCRUM en XP (zie verder) en merkten op dat hun methodes veel karakteristieken gemeen hadden. Hieruit ontstond het '*agile software development manifesto*' wat een symbolisch iets was maar wel getekend werd door alle zeventien aanwezigen. Zij noemen zichzelf de 'Agile Alliance', die dus bestaat uit een groep onafhankelijke denkers over softwareontwikkeling waarvan sommigen zelfs concurrenten zijn van elkaar (Beck et al., 2001).

Het *agile software development manifesto* laat zien dat er betere manieren zijn om software te ontwikkelen door in de praktijk aan te tonen dat dit werkt en door anderen ermee te helpen. Het verkiest volgens Beck et al.(2001):

- | | |
|--|-------------------------------------|
| - <u>mensen en hun onderlinge interactie</u> | boven processen en hulpmiddelen, |
| - <u>werkende software</u> | boven allesomvattende documentatie, |
| - <u>samenwerking met de klant</u> | boven contractonderhandelingen, en |
| - <u>inspelen op verandering</u> | boven het volgen van een plan. |

Beck et al. (2001) stellen dat er veel waardering is voor hetgeen aan de rechterkant vermeld staat, maar dat er meer waarde wordt gehecht aan wat er aan de linkerzijde genoemd wordt. Deze worden beschouwd als de kernwaarden van agile ontwikkeling. Elke individuele agile methode benadert deze kernwaarden op een enigszins andere manier, maar al deze methodes hebben specifieke processen en werkwijzen die een of meer van deze kernwaarden bevorderen (Sutherland, J., 2010).

De volgende twaalf leidende principes werden overeengekomen en zouden alle agile softwareontwikkelingsmethodes sturen in hun streven om flexibel en meer succesvol te zijn. Deze ondersteunen bovendien de bovenvermelde kernwaarden (Beck et al., 2001 & Sutherland, J., 2010).

- **Principe 1:** "De hoogste prioriteit is het tevredenstellen van de klant door het vroegtijdig en voortdurend opleveren van waardevolle software."
- **Principe 2:** "Verwelkom veranderende behoeftes, zelfs laat in het ontwikkelproces. Agile processen benutten verandering tot concurrentievoordeel van de klant."
- **Principe 3:** "Lever regelmatig werkende software op. Liefst iedere paar weken, hooguit iedere paar maanden."
- **Principe 4:** "Mensen uit de business en ontwikkelaars moeten dagelijks samenwerken gedurende het gehele project."

- **Principe 5:** "Bouwprojecten rond gemotiveerde individuen. Geef hen de omgeving en ondersteuning die ze nodig hebben en vertrouw erop dat ze de klus klaren."
- **Principe 6:** "De meest efficiënte en effectieve manier om informatie te delen in en met een ontwikkelteam is door met elkaar te praten."
- **Principe 7:** "Werkende software is de belangrijkste maat voor voortgang."
- **Principe 8:** "Agile processen bevorderen constante ontwikkeling. De opdrachtgevers, ontwikkelaars en gebruikers moeten een constant tempo eeuwig kunnen volhouden."
- **Principe 9:** "Voortdurende aandacht voor een hoge technische kwaliteit en voor een goed ontwerp versterken agility."
- **Principe 10:** "Eenvoud, de kunst van het maximaliseren van het werk dat niet gedaan wordt, is essentieel."
- **Principe 11:** "De beste architecturen, eisen en ontwerpen komen voort uit zelfsturende teams."
- **Principe 12:** "Op vaste tijden, onderzoekt het team hoe het effectiever kan worden en past vervolgens zijn gedrag daar op aan."

Om het gebruik van agile methodes te schetsen heeft onderzoek van het PMI aangetoond dat het gebruik van agile methodes in bedrijven verdrievoudigd is van december 2008 tot mei 2011 (Schwalbe, 2012). Volgens de VersionOne (2016) State of Agile Survey, is het zo dat 95% van de respondenten gebruik maken van agile methodes op de werkvloer en slechts 1% van de 3,880 respondenten vertelden dat de agile implementatie onsuccesvol was. Dit laat zien dat agile methodes tegenwoordig een onmisbaar iets zijn in de bedrijfswereld.

De meeste bekende agile methodes voor softwareontwikkeling zijn de volgende (Stoica et al. 2013):

- | | |
|--|---------------------------------------|
| - Extreme Programming (XP) | - Open unified process (OpenUP) |
| - Scrum | - Feature driven development (FDD) |
| - Dynamic Software Development Method (DSDM) | - Crystal |
| - Agile unified process (AUP) | - Agile modeling |
| - Essential unified process (EssUP) | - Adaptive Software Development (ASD) |

Elk van deze methodes zijn gebaseerd op de twaalf agile principes en heeft een unieke aanpak voor het ontwikkelen van goede software op een agile manier. Dit wil zeggen dat samenwerking wordt aangemoedigd en dat algemene beslissingen goed en snel genomen worden. Verder omvatten ze elk permanente communicatie, planning, testing en integratie. Deze methodes zijn gebaseerd op een iteratieve en incrementele ontwikkeling, waar specificaties en oplossingen voortvloeien uit samenwerkingen tussen individueel georganiseerde teams, maar met een gemeenschappelijk doel (Stoica et al., 2013).

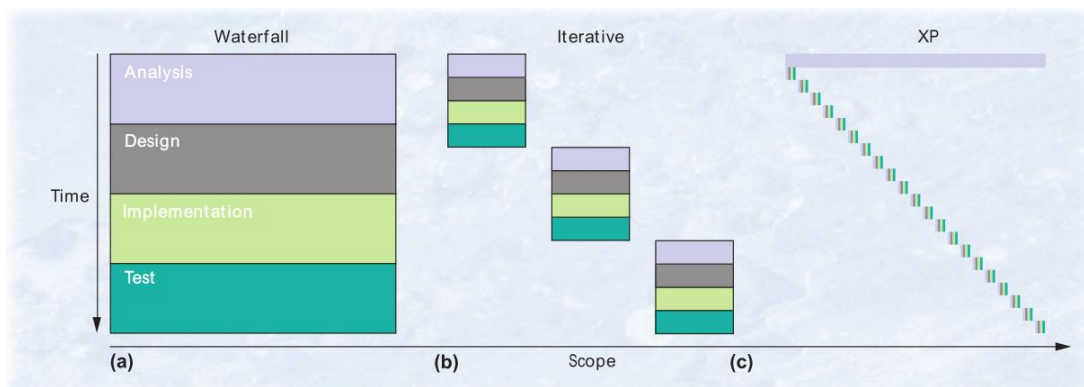
De focus van deze thesis ligt op de toepasbaarheid van agile projectmanagement in het hoger onderwijs en dus niet op de verschillende agile softwareontwikkelingsmethodes. Daarom zal de uitleg over de methodes beperkt worden tot Extreme Programming en Scrum. De reden waarom Extreme Programming uitgelegd zal worden is omdat dit een zeer bekende softwareontwikkelingsmethode is die in veel bronnen aan bod komt (zoals in: Melnik & Maurer, (2005); Rajamanickam, (2005); Beck, (1999); Maurer & Martel, (2002); Walczak, (2013); Stoica et al., (2013)) en de principes van agile

duidelijk in acht neemt. De reden waarom Scrum wordt aangehaald is omdat deze ook een zeer bekende agile methode die in veel bronnen over agile projectmanagement (zoals in: Ionel, (2008); Mahnic, (2010); Walczak, (2013); Valle & O'Mara, (2015); Stoica et al., (2013); Schwaber & Sutherland, (2016); May, York & Lending, (2016); Schwalbe, (2012); Jimenez & Cliburn, (2016)) aan bod komt en omdat deze richtlijnen, en dus geen softwareontwikkelingstechnieken, aan teams biedt. Dit leunt dus meer aan bij een algemene projectmanagementmethode. Verder is het zo dat Scrum in deze thesis nog verschillende keren aan bod zal komen aangezien dit een methode is die een poging tot implementatie kent in het onderwijs. Nog een reden waarom het beperkt is tot deze twee methodes, is omdat in de inleidende enquête die volgt in hoofdstuk 3, de respondenten voornamelijk kennis hadden van deze twee methodes.

2.4.1 Extreme Programming (XP)

Extreme programming is een modern ontwikkelingsmodel dat geïnspireerd is door RUP (Rational Unify Process). Het bestaat uit vier activiteiten, namelijk: coderen (hoofdactiviteit), testing (elke module die geprogrammeerd is moet getest worden), luisteren (de programmeur moet goed communiceren met de klant om te verstaan wat zijn noden zijn) en design (het bouwen van een juiste systeemarchitectuur zal leiden tot een efficiënt systeem en zal onnodige afhankelijkheden tussen modules verminderen). Het is geschikt voor projecten met dynamische benodigheden of projecten die niet goed gedefinieerd zijn in de startfase. De focus ligt hier op een goede samenwerking binnen het team wat wil zeggen dat ze worden aangemoedigd om hun persoonlijkheid te laten gelden alsook hun kennis uit te wisselen (Stoica et al., 2013).

Om het onderscheid te maken met de traditionele watervalmethode kan er gekeken worden naar figuur 5.



Figuur 6: Evolutie van de watervalmethode naar Extreme Programming (Beck, 1999)

Zo is er links op figuur 6 te zien dat de watervalmethode, zoals eerder vermeld, werkt met één cyclus waarbij de gebruiker precies vertelt wat hij of zij wil. Voor deze noden zal een systeem ontwikkeld worden die de gewenste features bevat. Het grote probleem hier is dat de gebruiker vaak niet weet hoe het systeem er precies van a tot z uit moet zien en er dus regelmatig veranderingen moeten plaatsvinden. Bij de watervalmethode wordt het veranderen van een deeltje van software duurder naarmate de tijd vordert. Het is dus best dat de grootste noden aan het begin van de cyclus worden ontdekt. Een eerste oplossing voor onverwachte veranderingen is te zien bij (b) op de figuur, waarbij er kortere ontwikkelingscyclussen worden gemaakt. Hier wordt er gewerkt met enkele iteraties

waarbij belangrijke veranderingen zouden kunnen worden geïmplementeerd. Een voorbeeld hiervan is het spiral model zoals kort uitgelegd wordt in (Awad, 2005). Doordat software projecten vaak in een steeds veranderende omgeving worden uitgevoerd is er nood aan een methode die kan omgaan met deze veranderende noden. Zo komen we tot figuur (c) wat de Extreme Programming methode voorstelt (Beck, 1999).

2.4.1.1 Aspecten van XP

XP bestaat uit zes verschillende aspecten, namelijk: de XP ontwikkelingscyclus, verhalen (stories), release, iteratie, taken en testen. De nadruk wordt, zoals eerder vermeld, gelegd op een nauwe samenwerking met de klant alsook de samenwerking binnen het team. Zo zullen er in een ontwikkelingscyclus de meest gewaardeerde en logisch bij elkaar horende features (dit wordt stories genoemd in XP en stellen de vereisten van het product voor), uit de lijst met alle features, door de klant gekozen worden. Hierbij zal er ook gekeken worden naar de kosten van de stories en de gemeten tijd voor het implementeren van de stories door het ontwikkelingsteam. Aan het begin van het project wordt samen met de klant bepaald wat het systeem zou moeten kunnen doen en wat er eerst moet gebeuren. Dit is te zien aan de lichtblauwe balk bovenaan op figuur 3 bij (c). Elke story moet business-georiënteerd, testbaar en meetbaar zijn. In een volgende iteratie zal de klant opnieuw de meest gewaardeerde stories kiezen uit de overblijvende lijst en dit blijft doorgaan tot alle features in het systeem verwerkt zijn. Elke story zal opgedeeld worden in kleinere taken waar de programmeurs individueel hun verantwoordelijkheid voor nemen. Wel moeten de programmeurs samen met een partner aan één computer werken (ook wel bekend als pair programming). Bij pair programming bedient één persoon de muis en het toetsenbord terwijl hij de code schrijft en denkt een andere persoon meer op strategisch vlak na om te bepalen of het de beste manier is om de functionaliteit te implementeren (Maurer, F. & Martel, S., 2002). Ze veranderen hun rollen enkele malen per dag. Dit bevordert de softwarekwaliteit alsook de jobtevredenheid van de programmeurs. Als er verder vragen zijn over de scope of aanpak van implementatie kan er samengezeten worden met de klant of met programmeurs die de meeste kennis hebben over het te schrijven deeltje code (Beck, 1999).

Als een taak voltooid is, zal de programmeur de taak in een set van test cases veranderen om aan te tonen dat een taak voltooid is. Deze zullen elk getest worden en moeten allemaal kunnen 'runnen' alvorens de delen van geschreven code samen geïntegreerd kunnen worden. Ook is het zo dat de programmeurs hun eigen tests schrijven alvorens ze code beginnen te schrijven. De samenhangende stories van een cyclus worden dan geïmplementeerd en zo zullen op het einde van het project al deze deeltjes samengebracht worden waardoor er één volledig werkend systeem ontstaat (Beck, 1999).

2.4.1.2 De onderliggende praktijken en filosofie van XP

Verder zijn er twaalf principes die de onderliggende praktijken en filosofie van XP voorstellen. Deze zijn ontworpen om efficiënter samen te werken en om de productiviteit te doen stijgen terwijl kwaliteit behouden wordt (Beck, K., 1999). Deze twaalf principes zijn onderverdeeld in drie belangrijke aspecten: klanttevredenheid, softwarekwaliteit en projectmanagement (Maurer & Martel, 2002).

Klanttevredenheid wordt pas bereikt wanneer het systeem de noden van de klant vervult. Dit wordt bij XP gedaan door een **on-site klant** en **kleine releases**. Zo is er met een on-site klant altijd iemand die bij een probleem input kan geven aan de programmeurs zodat de programmeurs niet hoeven te speculeren over wat de klant zou verkiezen. De kleine releases helpen om veranderende vereisten van de klant makkelijker te implementeren en reduceren de impact van planningsfouten. Verder waarborgt het dat elke release een nuttig softwaresysteemonderdeel is dat waarde voor de klant toevoegt (Maurer & Martel, 2002).

XP gebruikt verschillende praktijken om softwarekwaliteit hoog te houden. Het gecombineerd effect van deze verschillende praktijken verzekert dat het team hoge kwaliteit behoudt zonder dat het ontwikkelingsproces vertraagd wordt. Zo is er sprake van een **metafoor** waarin een samenhangend beeld van het systeem gerepresenteerd wordt dat logisch is vanuit het business en technisch perspectief en representeert "wat we proberen te doen". Dit wordt dus gedeeld tussen de klant en de programmeurs en dient als high-level softwarearchitectuur. Dan is er het **testgedeelte**, zoals al eerder werd vermeld, waarin een geautomatiseerde regressietest gebeurt. Deze functionele (aanvaardings)tests worden gedefinieerd door de klant en worden geïmplementeerd door de programmeur alvorens de softwarecode wordt geschreven. Verder probeert het team het **softwaredesign** zo **simpel** mogelijk te houden waardoor het team productiever kan werken, met minimale documentatie. Wanneer de broncode makkelijk te begrijpen is, is er geen nood om de structuur op een hoger niveau van abstractie te documenteren. Het volgende principe is **refactoring**, wat betekent dat de code geherstructureerd wordt om de onderhoudbaarheid en begrijpelijkheid te verbeteren zonder de functionaliteit te veranderen. Op deze manier blijft de broncode zo simpel mogelijk. Tot slot wordt er, zoals eerder vermeld en zeer belangrijk is, gebruik gemaakt van **pair programming**. (Maurer, F. & Martel, S., 2002).

XP bevat enkele praktijken om management overhead te reduceren terwijl de interesses van de klant op een gelijke hoogte worden gehouden. Zo gebruikt XP de **planning game** waarin de klanten de scope en de timing van de releases voorzien die gebaseerd zijn op schattingen die door de programmeurs zijn gemaakt. Om bijvoorbeeld inspanning te bepalen gebruikt XP *ideal engineering time (IET)*, wat de complexiteit en de snelheid van een taak meet. De snelheid wordt bepaald door eerdere ervaringen. Het tweede punt van projectmanagement is de **duurzame ontwikkeling** waarbij de XP filosofie zegt dat niemand meer dan 60 uur per week kan werken zonder dat dit invloed heeft op de kwaliteit van het product. Programmeurs moeten dus fris blijven om goede kwaliteit te leveren. Verder wordt er gesproken van **collectief eigendom** waarmee er wordt bedoeld dat de code eigendom is van iedereen in het team en dat iedereen die waarde kan toevoegen dit ook moet doen. **Continue integratie** is het volgende punt. Hierbij wordt er gezegd dat programmeurs de ontwikkelde code zo vaak mogelijk moeten integreren en liefst eenmaal per dag. Zo verzekeren ze dat er altijd een uitvoerbare systeemversie beschikbaar is die alle nieuwe functies bevat en kan dienen als basis voor al het werk (Maurer & Martel, 2002).

Tot slot verschilt het 12de principe vermeld door Maurer & Martel (2002) van dat van Beck (1999). Maurer & Martel spreken over codestandaarden omdat er verschillende teams aan verschillende delen van het systeem werken waardoor een codestandaard een must is. Beck daarentegen, spreekt over een open werkplek waar de teams in een grote ruimte werken met kleine cabines rond de periferie

en waarbij de paar programmeurs op computers werken die opgesteld staan in het centrum. Deze 12 principes zijn regels die je zou moeten volgen. Maar het zijn slechts regels en hiervan kan altijd afgeweken worden. Het niet implementeren van alle regels kan er wel voor zorgen dat je niet kan genieten van alle potentiële voordelen.

2.4.2 SCRUM

Een andere methode die vaak gehanteerd wordt, is Scrum. Stoica et al. (2013) definieert Scrum als een iteratieve en incrementele methode waarvan het doel is om ontwikkelingsteams te helpen zich te concentreren op het behalen van doelen en het werk van minder belangrijke taken te minimaliseren. Zo probeert Scrum eenvoud te behouden in een ingewikkelde business omgeving door richtlijnen, dus geen softwareontwikkelingstechnieken, aan teams te bieden. Deze richtlijnen focussen zich op de interactie tussen teamleden om zo een flexibel, aangepast en productief systeem te creëren in een steeds veranderende omgeving.

Schwaber en Sutherland (2016) definiëren Scrum als een framework waarin mensen moeilijke en veranderende problemen kunnen aanpakken terwijl ze op een productieve en creatieve manier producten van de hoogst mogelijke waarde afleveren.

Uit deze twee definities kan er worden afgeleid dat Scrum ervoor zorgt dat er wordt gefocust op de belangrijkste zaken op een iteratieve en incrementele manier. Hierbij is er veel interactie, creativiteit en flexibiliteit tussen de teamleden nodig om zo een product van hoge kwaliteit en waarde te kunnen leveren.

Hetgeen SCRUM doet verschillen van andere agile methodes, is dat het niet specifiek software gerelateerd is. Deze methode kan in verschillende situaties worden toegepast en voor verschillende soorten projecten worden gebruikt (Jimenez & Cliburn, 2016).

2.4.2.1 SCRUM rollen

Een team dat gebruik maakt van Scrum specificeert rollen, artefacten en meetings. Er zijn drie hoofdrollen voor de deelnemers van het project, namelijk: de Product Owner, de ScrumMaster en het Scrum team. De **Product Owner** is de verantwoordelijke voor de business waarde van het project en de volgorde waarin het werk gedaan moet worden (zoals aangegeven in de product backlog). Meestal wordt het project in het leven geroepen door een persoon met grote verantwoordelijkheden en een hoge functie zoals een CEO. Deze wordt dan ook in de meeste gevallen beschouwd als de product owner (Schwalbe, 2012). De focus van de product owner is het verzekeren dat de items die in de product backlog staan effectief overeenkomen met de noden van de klant op het desbetreffende moment. Dit in tegenstelling tot de traditionele projectmanager, waarvan de focus ligt op het behouden van de balans tussen functionaliteit, tijd en kosten (May, York & Lending, 2016).

De tweede rol is de **ScrumMaster** rol. Deze persoon heeft als verantwoordelijkheid het voorkomen en oplossen van belemmeringen die ervoor zouden kunnen zorgen dat het team niet optimaal en productief kan werken (May et al., 2016). Verder vergemakkelijkt de ScrumMaster de daily Scrum meetings en zorgt hij/zij ervoor dat alle rollen en functies goed op elkaar zijn afgesteld. Het is ook zo dat de ScrumMaster, in tegenstelling tot een traditionele projectmanager, geen gezag heeft over

de teamleden maar wel over het proces. Hij/zij verzekert dus dat het team alle benodigheden heeft om het werk gedaan te krijgen (Schwalbe, 2012).

Als derde rol is er het **ontwikkelingsteam** dat in het bedrijfsleven een cross-functioneel team is dat bestaat uit vijf tot negen zelforganiserende personen die het gewenste product elke sprint leveren. Zo een sprint duurt gewoonlijk tussen de twee à vier weken waarin een specifiek deel van de product backlog moet worden voltooid en beoordeeld (Schwalbe, 2012). De ideale team grootte wordt in verschillende literatuur anders voorgesteld. Volgens Ionel (2008) ligt de ideale grootte tussen één en tien personen. Deze auteur benadrukt dat dit aantal niet overschreden mag worden omdat kleine teams zelfstandiger en efficiënter werken. Zo is de communicatie bij kleinere projecten meestal gemakkelijker zodat iedereen altijd up-to-date is met de vooruitgang van het project. Ook kan er op deze manier makkelijker met de klant afgesproken worden en is er een kans dat elk teamlid met de klant kan samenzitten. Dit zorgt ervoor dat de interactie tussen de teamleden en de klant bevordert wordt en de teamleden precies weten wat ze moeten doen. In figuur 7 wordt de typische werking van een scrumproject getoond.



Figuur 7: het Scrum framework (May et al., 2016)

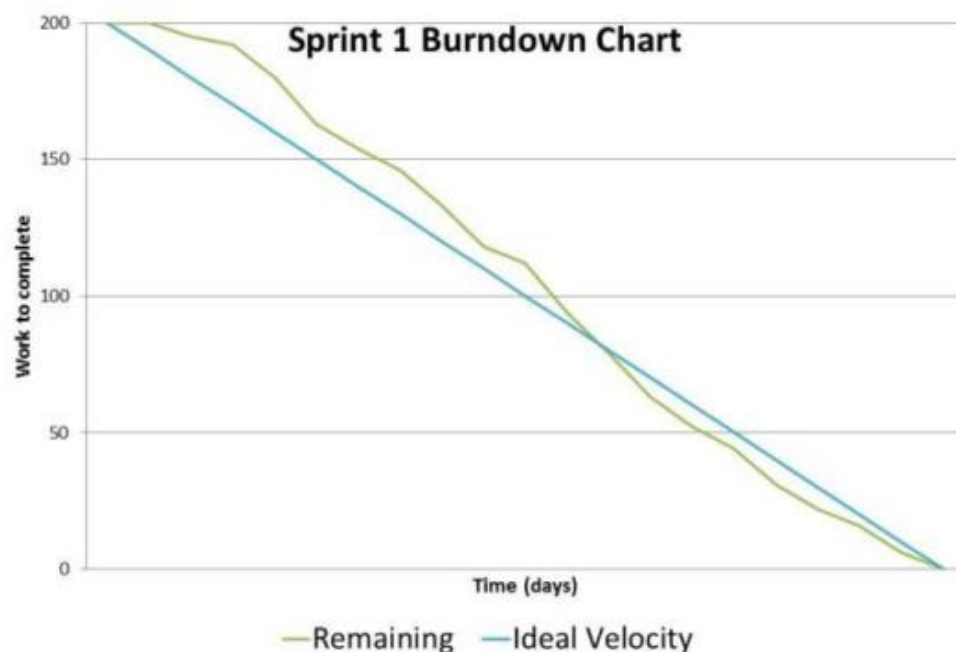
2.4.2.2 SCRUM artefacten

In figuur 7 is te zien dat de product owner nadenkt over features die het product zeker moet hebben. Deze worden allemaal in de **product backlog** geplaatst. Dit wordt in Scrum als een artefact beschouwd, wat een nuttig object is, gecreëerd is door mensen. De product backlog is dus een geordende lijst met features waarin prioriteiten op basis van business value worden gesteld (Schwalbe, 2012). In tegenstelling tot een traditioneel project, waarbij er een vaste set van vereisten aan het begin van het project wordt gespecificeerd, is de product backlog een dynamische lijst die steeds verandert tot de noden die het project vereisen duidelijk zijn voor de klant (May et al., 2016).

Naast de product backlog is er de **sprint backlog** waarbij de items met de hoogste prioriteit van de product backlog in de desbetreffende sprint voltooid moeten worden. Dit valt te vergelijken met het selecteren van de belangrijkste stories bij XP. Zo zullen ook hier de items in kleinere taken worden

verdeeld en door de teamleden voltooid. Schwalbe (2012) beweert dat de teamleden ongeveer zestien uur de tijd hebben om deze taken te voltooien.

Verder wordt er vaak gebruik gemaakt van **Burndown charts**, wat ook beschouwd wordt als een artefact. Dit meet het cumulatieve werk (de som van de tijd die nodig is om de verschillende taken te voltooien) dat er nog verricht moet worden in functie van tijd (meestal in dagen). Zo is in figuur 8 de ideale lijn te zien in het blauw en de werkelijke lijn in het grijs (Schwalbe, 2012). Op deze figuur leunt het werkelijke nauw aan bij het ideale en is de lijn strikt dalend, terwijl dit in de realiteit niet altijd zo zal zijn. Het kan zijn dat de werkhoeveelheid terug stijgt wanneer er bijvoorbeeld onverwachte veranderingen moeten worden doorgevoerd die extra tijd met zich meebrengen. Burndown charts kunnen met excel worden gemaakt, maar hier bestaan ook speciale tools voor. Meer uitleg over de tools die gebruikt worden bij agile (alsook traditionele) softwareontwikkeling methodes kan je terugvinden in Dubakov & Stevens (2008).



Figuur 8: Voorbeeld: Burndown chart (Schwalbe, 2012)

2.4.2.3 SCRUM meetings

Er bestaan vier meetings die worden georganiseerd en begeleid door de ScrumMaster. Zo is er de **Sprint planning** sessie, waarbij het ontwikkelingsteam, de ScrumMaster en de product owner samenwerken om te bepalen welke items uit de product backlog gedurende de desbetreffende sprint zullen behandeld worden. Dan is er de **Daily Scrum** wat een korte meeting aan het begin van de dag voorstelt. Hier wordt de vooruitgang gedeeld tussen de teamleden van de voorbije dag, bespreken ze de uitdagingen die er hen die dag te wachten staan en worden belemmeringen waardoor ze het werk niet gedaan krijgen gecommuniceerd. Dit neemt ongeveer een kwartier in beslag. Gedurende de **sprint review**, wat aan het einde van een sprint gehouden wordt, gaat het team aan de product owner tonen wat ze voltooid hebben gedurende de sprint. Hier heeft de klant de kans om het product te zien en feedback te voorzien op het productincrement. Tot slot is er de

sprint retrospective, waar het team gaat discussiëren over wat er goed werkte en waar het misschien beter kan. Op deze manier kunnen ze verbeteringen in het proces snel analyseren en implementeren (Schwalbe, 2012; May et al., 2016).

2.5 Agile projectmanagement in het algemeen

Wat duidelijk is uit het voorgaande is dat agile projectmanagement een managementmethodologie is die voornamelijk van toepassing is in een software gerelateerde context. Om verder niet specifiek in te gaan op één bepaalde agile methode, wordt hier agile projectmanagement in het algemeen toegelicht.

Agile projectmanagement wordt door Hass (2007) gedefinieerd als een hoog iteratief en incrementeel proces, waar de ontwikkelaars en project stakeholders actief samenwerken om het domein te begrijpen, identificeren wat er gebouwd moet worden en de prioriteit geven aan de functionaliteit. Agile methodes kunnen gebruikt worden wanneer de volgende voorwaarden zich voordoen (Hass, 2007):

- de waarde van het project is duidelijk;
- actieve participanten gedurende het project;
- de klant, designers en ontwikkelaars zijn mede gelokaliseerd;
- incrementele feature-driven development is mogelijk (dit wordt verder kort beschreven); en
- visuele documentatie (Kaarten op de muur vs. Formele documentatie) is accepteerbaar.

De belangrijkste agile concepten zijn volgens Karlesky & Van Der Voord (2008) het definiëren van de klant, de features en 'done'. Dit laatste betekent dat er moet gedefinieerd worden wanneer een feature als klaar beschouwd kan worden. Uit het voorgaande was al duidelijk dat de klant zo dicht mogelijk bij het ontwikkelingsteam moet staan zodat de keuzes in de juiste richting genomen kunnen worden, de prioriteiten van de features beter bepaald kunnen worden en dat er vragen over het domein beantwoord kunnen worden. Zo is het ook dat een feature gedefinieerd wordt vanuit het perspectief van de klant. Deze wordt met eigen woorden van de klant beschreven, geverifieerd door de klant wanneer de feature klaar is en moet genoeg waarde creëren zodat de klant bereid is ervoor te betalen. Een feature wordt als done beschouwd wanneer het op een uitgebreide manier getest kan worden. Zo moet het de unit-, systeem- en acceptatietest doorstaan. Zoals aangehaald bij de XP methode, is het vaak zo dat de unit tests en systeem tests op voorhand door de ontwikkelaars (zoals bij Test-Driven Development) of parallel met het schrijven van de code gecreëerd worden. Ook zijn deze tests geautomatiseerd zodat deze uitgevoerd kunnen worden als een regressie-testsuite. Een regressie-testsuite bestaat uit een reeks testscenario's die zijn ontworpen om ervoor te zorgen dat de software nauwkeurig en correct is na het ondergaan van correcties of wijzigingen. De acceptatie tests worden gedaan door de klant en kunnen op verschillende manieren gebeuren (Karlesky & Van Der Voord, 2008).

De volgende elementen zijn volgens Hass (2007) de belangrijkste elementen die de basis voor agile projectmanagement bieden. De meeste van deze elementen zijn al aan bod gekomen en worden dus niet meer uitgelegd, maar worden hier wel opgesomd om een algemeen beeld over agile projectmanagement te krijgen. Ook kunnen ze toegepast worden door de traditionele methodes om de projectprestatie te verbeteren (Hass, 2007).

1. **Visuele controle:** Bij APM wordt er vaak gebruik gemaakt van de zogenaamde "cards-on-the-wall" methode van planning die het team helpt om het werk van het project te organiseren. Deze methode bestaat ook in de vorm van tools.
2. **Co-located high-performing teams**
3. **Test-driven development**
4. **Adaptieve controle:** Agile teamleden passen zich voortdurend aan om hun methodes te verbeteren. Na elke iteratie gaan ze de "lessons learned" implementeren in de volgende iteratie.
5. **Collaboratieve ontwikkeling**
6. **Feature-driven development:** Het team focust zich telkens op één feature per keer. Dit zorgt voor een daling in complexiteit.
7. **Leiderschap en collaboratie in plaats van command en control**
8. **De focus veranderen van kosten naar opbrengsten:** Features worden volgens prioriteit gebaseerd op waarde, zoals verhoogde opbrengsten of marktaandeel. Zo moet de business analist verzekeren dat er niet te veel wordt geïnvesteerd in het ontwikkelen van een nieuwe oplossing. De projectmanager focust zich op de projectkosten en de business analist focust zich op de TCO (total cost of ownership). Dit laatste bevat de ontwikkelingskost en de kosten van het bedienen van het nieuwe systeem nadat het geïmplementeerd is.
9. **Lessons learned:** Dit is vergelijkbaar met de sprint retrospective bij Scrum waarbij er na elke sprint wordt samengezeten om te kijken waar het beter kan. Mogelijke verbeteringen worden geïmplementeerd in de volgende cyclus.

Verder zijn er volgens Karlesky (2008) nog enkele andere elementen die zeker belangrijk zijn bij agile projectmanagement:

- **Simpliciteit:** Onnodige complexiteit zijn onnodige kosten. Zo is er door complexiteit meer kans op fouten, kans op moeilijkheden bij het testen, kans op trage vooruitgang en het verdoezelt code.
- **Functionaliteit:** In de realiteit worden vaak componenten geprogrammeerd die nooit worden gebruikt. Bij APM wordt ervan uit gegaan dat enkel de features worden ontwikkeld die ook werkelijk nodig zijn. De ontwikkelaars vertrouwen op de suites van regressietests en kunnen zo, zoals eerder vermeld bij XP, bestaande code refactoren en functionaliteit toevoegen wanneer het werkelijk nodig is in latere stadia van het project.
- **Documentatie:** APM heeft een voorkeur voor beperkte, flexibele en just-in-time documentatie zoals ook al werd aangehaald bij XP. Finale documentatie moet simpel zijn en moet high-level overzichten hebben van belangrijke features en subsystemen.
- **Continue integratie:** Dit zorgt ervoor dat de systeem code-base altijd grondig wordt getest en dat er geen problemen zijn met integratie van subsystemen of secties van code
- **Risico management en Scope management:** De meest risicovolle zaken van het project krijgen prioriteit op de rest en moeten dus als eerste voltooid worden. Zo wordt de burndown chart na elke iteratie geupdate en worden er beslissingen gemaakt over de bestaande en nieuwe features
- **Planning poker:** Dit is ook aan bod gekomen bij XP als de planning game. Een groot probleem bij software projecten is het schatten van hoelang het duurt om een feature klaar

te krijgen aangezien de complexiteit tussen de features vaak verschilt. Om dit op te lossen wordt er een complexiteitspunt aan de features toegewezen voordat deze in de product backlog terecht komt

- **Snelheid (Velocity):** Velocity of snelheid is een simpele ratio van complexiteitspunten voltooid per iteratie. Als er een feature als 'done' wordt beschouwd, worden de complexiteitspunten van deze feature toegewezen aan de iteratie waarin het zich bevond. Aan de hand van deze manier kan de tijd die nodig is voor het voltooien van het project geschat worden. Dit wordt berekend door het delen van de backlog van complexiteitspunten door de snelheid. Het resultaat van deze ratio is het aantal overblijvende iteraties die nog voltooid moeten worden. Zo kunnen er vroegtijdig aanpassingen doorgevoerd worden bij de middelen, features en prioriteiten wanneer er opgemerkt wordt dat de voltooiing van het project de budget limiet of de bezorgdatum zou overschrijden.

2.5.1 Risico's verbonden aan APM met bijhorende reacties

Agile projectmanagement zorgt ervoor dat er risico's die zich voordoen bij de traditionele methodes teniet gaan. Er wordt vanuit gegaan dat expliciet risicomangement niet nodig is aangezien de APM methodologie ontworpen is om hoge risico projecten te voltooien. Dit wil echter niet zeggen dat er aan agile projectmanagement geen risico's verbonden zijn. De auteurs Walczak en Kuchta (2013) lichten verschillende risico's toe die gepaard gaan met het gebruiken van agile projectmanagement. De risico's die er zijn gedetecteerd zijn gebaseerd op risico's die zich afspelen bij het gebruik van scrum, XP en de APM methode (hier wordt de APM methode als een aparte methode beschouwd) (Walczak & Kuchta, 2013). Aangezien het verdere verloop van de thesis over het toepassen van agile projectmanagement in de hoger onderwijscontext zal gaan, zullen hier de belangrijkste risico's met de bijhorende 'oplossingen' die ook toepasbaar zijn in het onderwijs beschreven worden. Dit betekent dat de focus hier niet zal liggen op het softwareontwikkelingsaspect, maar eerder op de risico's die er zich kunnen voordoen in teams wanneer er volgens agile te werk gegaan wordt. Meer informatie omtrent de risico's en oppurtiniteiten van het toepassen van APM kan er gevonden worden in (Walczak & Kuchta, 2013).

2.5.1.1 Development process risks

De agile methodologie wordt beschouwd als een methodologie voor zowel softwareontwikkeling als projectmanagement. De risico's van het ontwikkelingsproces verwijzen hier naar de risico's verbonden aan de agile methodologie die er gebruikt wordt. De risico's die veroorzaakt worden door de methodologie omtrent ontwikkelingsproces zullen hier toegelicht worden (Walczak & Kuchta, 2013):

- **Inefficiënte SCRUM vergaderingen en ineffectieve SCRUM rollen:** Deze twee risico's kunnen resulteren in projectvertragingen en kostenoverschrijdingen. Er zijn twee methodes om deze risico's te verkleinen. Ten eerste kan het proces aangepast worden om de hoeveelheid tijd besteed aan onproductieve taken of vergaderingen te verminderen en worden de rollen geoptimaliseerd. Ten tweede kan er gewerkt worden met vaste regels

(bijvoorbeeld: vaste agenda's voor alle Scrum meetings en time-boxing³), die het team voorzien van training en coaching over de agile methodes. Tot slot is er nog een risico omtrent het toewijzen van Scrum rollen. De ScrumMaster moet ervaren zijn of zijn vaardigheden verbeteren. Hij/zij zou anders mogelijk te veel tijd verbruiken aan zijn eigen rol, maar ook de onproductieve tijd van het team vergroten.

- **Het team is niet in staat om zichzelf te organiseren en groepsbeslissingen te nemen:** Voorbeelden hiervan zijn:
 - o Het team is niet in staat zichzelf te organiseren door interne conflicten. Dit voorkomt om tot een consensus te komen.
 - o Het team is overweldigd door de grote hoeveelheid informatie die verwerkt moet worden en het aantal beslissingen die genomen moeten worden. Verder kan het zijn dat de specialisaties van de teamleden niet breed genoeg zijn, waardoor ze 'the big picture' niet begrijpen.

Een oplossing zou zijn om enkele teamleden te verplaatsen, maar de beste oplossing is om de teamdynamiek te verbeteren. Dit kan gedaan worden door team building activiteiten aan het begin van een project, het geven van grondige trainingen in agile methodes, het aannemen van een ervaren ScrumMaster, agile coaching gedurende de eerste sprints en het beperken van het zichzelf organiseren door middel van basisregels die de teamleden moeten volgen.

- **Foutieve groepsbeslissingen:** Bij agile methodes wordt ervan uit gegaan dat de groepsbeslissingen die genomen worden door een team beter zijn dan de beslissingen die individueel genomen worden. De mogelijkheid bestaat steeds dat het tegenovergestelde gebeurt. Aangezien het team, bij agile methodes, ook leidinggevende verantwoordelijkheden krijgt, moeten de teamleden ervoor zorgen dat ze naast de technische competenties ook een end-to-end begrip van het hele project hebben. Op deze manier worden de juiste beslissingen genomen.
- **De teamleden hebben onvoldoende kennis van de agile methodes:** Het komt weleens voor dat teamleden verschillende interpretaties hebben over de agile methode die ze toepassen, waardoor ze de methode op een foutieve manier toepassen. Op deze manier geraken ze in chaotische werksituaties en zal de moreel in het team laag zijn. Het is dus heel belangrijk dat de werknemers getraind en ondersteund worden door een agile coach die hun laat zien hoe het beter kan. Takpuie & Maureen (2016) hebben onderzoek gedaan naar de karakteristieken die Scrum teamleden nodig hebben om op een succesvolle manier kennis over te dragen naar anderen. Uit hun studie bleek dat intrinsieke motivatie een grotere invloed op kennisoverdracht heeft dan extrinsieke motivatie. Verder was bekwaamheid van de teamleden ook een belangrijk punt, maar dit alleen in combinatie met motivatie. Geloofwaardigheid, zoals reputatie en ervaring van de teamleden of scrum master, heeft ook

³ Time-boxing kent een vaste, maximale tijdseenheid voor een activiteit. Die tijdseenheid wordt een time box genoemd. Het doel van timeboxing is om de hoeveelheid tijd die aan een activiteit is toegewezen te definiëren en te beperken. Scrum gebruikt timeboxing voor alle Scrum-evenementen en als een hulpmiddel voor het concreet definiëren van open of dubbelzinnige taken. <https://www.scruminc.com/what-is-timeboxing/>

een positieve invloed op de overdracht van kennis. Ook duidelijke en beknopte communicatie en zelfs empathie spelen een rol bij de kennisoverdracht.

2.5.1.2 Management process risks

Zoals vermeld bij ontwikkelingsproces risico, wordt de agile methodologie behandeld als een projectmanagement methodologie. De bron van de risico's zit in de methodologie zelf en worden hierna beschreven (Walczak & Kuchta, 2013):

- **Foutieve implementatie van de artefacten:** De agile methodes zorgen ervoor dat er een betere projectmonitoring en -controle kan gedaan worden ten opzichte van andere methodes. Dit omwille van de hoge visibiliteit van de staat van het project op elk gegeven moment aangezien er tools zijn die deze kunnen visualiseren, zoals de burndown chart. Ook is de staat van het project beter zichtbaar door de korte cycli die de agile methodes hanteren. Op deze manier kan de projectmanager te weten komen als er zich beperkingen voordoen waardoor bepaalde projectmijlpalen niet bereikt kunnen worden en heeft hij/zij genoeg tijd om hierop te reageren. Er doet zich een probleem voor wanneer de agile artefacten zoals de burndown charts of de product backlog niet op een fatsoenlijke wijze zijn geïmplementeerd of worden geüpdatet. Om dit op te lossen moet er optimaal gebruik gemaakt worden van deze artefacten en moeten deze voortdurend up-to-date zijn met betrouwbare en accurate informatie.
- **De methodologie van projectmanagement weegt op tegen individuele carrièreplannen:** De traditionele rollen van projectleiders of teamleiders bestaan niet binnen de agile methodologie. Dit kan botsen met werknemers die willen doorgroeien in een bedrijf en carrière willen maken. De mensen die een agile team betreden zouden zo vroeg mogelijk geïnformeerd moeten worden over de structuur van het team. Wanneer deze mensen dit niet zien zitten moet er gekeken worden naar een herpositionering, want in het slechtste geval kan een bedrijf waardevolle en eraven personen verliezen.

2.5.1.3 Work environment risks

Risico's voor een agile project kunnen ook komen van de werkomgeving waar het project wordt uitgevoerd. Als de omgeving zich niet heeft aangepast aan de agile methodologie, dan is dit een grote bron van risico's voor het project (Walczak & Kuchta, 2013):

- **De organisatie volgt de regels van agile niet:** Dit is een typisch voorbeeld van wat er in het dagelijks leven gebeurt. Elke organisatie heeft moeilijkheden of komt obstakels tegen bij het implementeren van de agile methodologie. Het grootste obstakel is dat de werknemers zich op een traditionele manier blijven gedragen. Hiermee wordt bedoeld dat er bijvoorbeeld toch met een leider gewerkt wordt, dat er verticale communicatie is, dat er werknemers instaan voor verschillende projecten, enzovoort. Een goede oplossing hiervoor is het gebruik van de agile methodes op organisatorisch niveau in plaats van alleen op projectniveau. Natuurlijk moeten de agile methodes eerst geïmplementeerd worden op projectniveau aangezien het langer duurt om het te implementeren op organisatorisch niveau.

2.5.1.4 Customer risks

Een laatste belangrijk punt zijn de risico's verbonden aan de samenwerking met de klant. Volgens de agile methodologie moet er een nauwe samenwerking zijn met de klant bij elke staat van het project. Hieronder worden twee van de veelvoorkomende oorzaken beschreven (Walczak & Kuchta, 2013):

- **De klant handelt op een traditionele manier:** Zo zal de klant heel actief zijn in het begin van het project wanneer het contract getekend wordt en de vereisten gespecificeerd worden. De klant zal ook actief zijn aan het einde van het project wanneer hij/zij de project deliverables moet goedkeuren. De klant toont weinig interesse in het volledige verloop van het project. Het grote probleem hier is dat wanneer de assumpties aan het begin van het project fout of onvolledig zouden zijn, het eindresultaat niet aan de wensen van de klant voldoet. De klant zou dus aan het begin van het project geïnformeerd moeten worden over het gewenste niveau van interactie met de organisatie.
- **De klant is niet in staat om op tijd waardevolle informatie te geven:** De grootte en de complexiteit van de organisatie van de klant en de weinige interesse in het verloop van het project, kan ervoor zorgen dat de klant geen waardevolle en volledige informatie op tijd kan geven. Zo zal de klant ook aan het begin moeten worden geïnformeerd over het gewenste niveau van communicatie tussen de klant en de organisatie.

2.6 De algemene verschillen tussen traditionele en agile projectmanagement methodes

Om alle elementen van de traditionele en agile projectmanagement methodes in kaart te brengen, kunnen de verschillen tussen de traditionele en agile methodes teruggevonden worden in tabel 1. Hier wordt de focus eerder gelegd op het managementaspect in plaats van het softwareontwikkelingsaspect, maar sommige elementen van softwareontwikkeling zijn cruciaal en moeten hier ook beschreven worden (Stoica et al., 2013; Fernandez & Fernandez, 2008).

	Traditioneel	Agile
Projecten	<ul style="list-style-type: none">- Duidelijk omschreven- Uitgebreid gedocumenteerde features, functies en stabiele vereisten- Grotere projecten	<ul style="list-style-type: none">- Vereisten worden duidelijker na elke iteratie en vermindert/elimineert onzekerheid- Flexibel: makkelijker inspelen op veranderingen van vereisten- Kleinere projecten
Projectmanagers	<ul style="list-style-type: none">- Managen het project op basis van budget, planning en scope- Willen risico verminderen en beperkingen van tijd en geld behouden (hoge veiligheid)	<ul style="list-style-type: none">- Focus ligt op het snel leveren van producten en business waarde (budget en tijdlijn zijn secundair)

Projectteams	<ul style="list-style-type: none"> - Makkelijker ondersteunen van teams (door goed gedefinieerde vereisten en documentatie) - Grote teams 	<ul style="list-style-type: none"> - Nood aan het werken op dezelfde locatie van de teamleden en personeel zodat er makkelijk ingespeeld kan worden op veranderingen en makkelijker productonderdelen kunnen worden geproduceerd - Goede samenwerking - Geavanceerde kennis - Kleine teams
Managementstijl	<ul style="list-style-type: none"> - Command and control 	<ul style="list-style-type: none"> - Leiderschap en collaboratie
Kennismanagement	<ul style="list-style-type: none"> - Expliciet 	<ul style="list-style-type: none"> - Stilzwijgend
Communicatie	<ul style="list-style-type: none"> - Formeel 	<ul style="list-style-type: none"> - Informeel
Organisatiestructuur	<ul style="list-style-type: none"> - Mechanisch (bureaucratisch, hoge formalisering), gericht op grote organisaties 	<ul style="list-style-type: none"> - Organisch (flexibel en deelnemend, moedigt sociale samenwerking aan), gericht op kleine en middelgrote organisaties
Kwaliteitscontrole en testing	<ul style="list-style-type: none"> - Moeilijke planning en strikte controle. Moeilijke en late testing (nadat code is geïmplementeerd) 	<ul style="list-style-type: none"> - Permanente controle en permanente testing (na elke iteratie)
Kost van herstart	<ul style="list-style-type: none"> - Hoog 	<ul style="list-style-type: none"> - Laag
Additionele vaardigheden van de ontwikkelaars	<ul style="list-style-type: none"> - Niets specifiek 	<ul style="list-style-type: none"> - Interpersoonlijke vaardigheden en basiskennis van het bedrijf

Tabel 1: De verschillen tussen traditionele en agile PM (Stoica et al., 2013) (Fernandez & Fernandez, 2008)

2.6.1 De verschillende principes waarnaar moet gekeken worden bij het kiezen van een projectmanagementmethodologie

Naast de twaalf principes en waarden van het agile manifesto die eerder werden beschreven heeft Cockburn (2000) nog vier andere principes aangehaald die bekeken moeten worden bij het selecteren van een projectmanagementmethodologie. Deze worden hieronder beschreven:

1. Een grotere groep heeft nood aan een grotere methodologie: een methodologie is groter wanneer het meer elementen bevat (rollen, standaarden, werkproducten, ...).
2. Een kritischer systeem, een systeem waarvan de ongedetecteerde effecten meer schade verrichten, heeft meer openbare zichtbare correctheid (grotere dichtheid) nodig bij de constructie.

3. Een relatief kleine toename in de grootte of dichtheid van een methodologie voegt een relatief groot bedrag toe aan de projectkosten.
4. De meest effectieve vorm van communicatie is interactieve en face-to-face communicatie.

Verder zijn projectprioriteiten, bezorgdatum, kwaliteitsniveau en de gewenste visibiliteit van het proces ook elementen die de projectaanpak kunnen beïnvloeden. Er moet ook een match bestaan tussen het project, de cultuur van het bedrijf, de klanten en de projectstrategie die geselecteerd wordt (Fernandez, D.J. & Fernandez, J.D. 2008).

Hieruit is het duidelijk dat de keuze van een projectmanagement methodologie niet alleen afhangt van welke methodologie een bedrijf wil toepassen, maar ook van hoe de organisatie er uitziet, hoe deze werkt, hoe groot ze is, wat de waarde en normen zijn, enzovoort. Een organisatie kan zich helemaal aanpassen om agile te worden, maar bij grote organisaties zullen er altijd struikelblokken zijn die het agile aspect niet volledig toelaten.

Bedrijven kunnen wel onderdelen van de agile projectmanagementmethodologie implementeren in de organisatie. In de literatuur wordt er gesproken van hybride aanpakken waarin zowel traditionele als agile praktijken gebruikt worden (Fernandez & Fernandez, 2008). Volgens Rico (2008) zijn hybride methodes de laatste trend waarin zowel verschillende agile methodes zoals XP en Scrum gecombineerd worden, maar ook traditionele methodes met agile methodes een mix kent.

Rico (2008) heeft een studie gedaan waarbij de kosten en baten van de agile methodes met elkaar vergeleken werden. Hieruit bleek dat de agile methodes gemiddeld 200% beter presteren dan de grote en dure traditionele methodes. In tabel 1 werd verteld dat agile methodes bedoeld zijn voor kleine bedrijven en kleine projecten. Dit komt doordat 70% van de kleine tot middelgrote projecten de agile methodes gebruiken. Rico (2008) beweert dat voor grote en complexe projecten de relevantie van agile methodes nog moet worden bewezen. Hiernaar zou verder onderzoek gedaan moeten worden.

Tot slot zijn er ook uitbreidingen van de Agile methodes die recent in het leven zijn geroepen, DevOps (Development & Operations) is hier een voorbeeld van. DevOps wordt in Banica et al. (2017) gedefinieerd als: "Een softwareontwikkelingsmethodologie die kijkt naar het integreren van alle softwareontwikkelingsfuncties, van ontwikkeling tot operaties, in dezelfde cyclus". Deze methode is in het leven geroepen omdat er ontdekt is dat de scheiding tussen softwareontwikkeling en softwareoperaties ervoor zouden kunnen zorgen dat er een vertraging is op het vinden van fouten. Dit zorgt voor een shift van de deadlines van projecten. DevOps tracht deze twee componenten te harmoniseren. Meer informatie over DevOps kan je terugvinden in Banica et al. (2017) en op de website van VersionOne (www.versionone.com).

2.7 Agile projectmanagement in het hoger onderwijs

Uit het vorige is duidelijk dat de agile methodologie voornamelijk gebruikt wordt in de bedrijfswereld. Het hoger onderwijs kent recent ook aanrakingen met de agile aanpak. Afgeleid uit de jaartallen van uitgave van de literatuur kent het toepassen van agile methodes in het hoger onderwijs een opmars sinds 2015. Het betreft voornamelijk het toepassen van Scrum in opleidingsonderdelen rond softwareontwikkeling. Ook is er in 2015 een gids, genaamd eduScrum, geschreven door Delhij, Van

Solingen, en Wijnands (2015) die voornamelijk gericht is op het secundair onderwijs maar een aangepaste versie van de originele SCRUM methode voorstelt. EduScrum wordt gedefinieerd als: "Een kader waarbinnen studenten complexe aanpassingsproblemen kunnen aanpakken, terwijl ze productief en creatief leerdoelstellingen en persoonlijke groei van de hoogst mogelijke waarde bereiken". De eduScrum gids vertelt je dus wat je moet doen om volgens eduScrum te werken, maar vertelt je niet hoe je dit moet doen. In de eduScrum gids staat leren centraal: leer slimmer, verbeter collaboratie en leer jezelf beter kennen. Meer informatie is terug te vinden in de eduScrum gids (Delhij et al., 2015).

Een studie die gedaan werd door Rico en Sayani (2009) benadert het gebruik van verschillende agile methodes in een project van een Capstone course (een finaal vak dat nodig is voor het behalen van een bachelor of master diploma: hier betreft het het behalen van een master diploma in Software Engineering). De conclusie die werd bekomen is dat teamwerk en samenwerking met de klant de basiswaarden van de agile methodes zijn en dat deze bij traditionele methodes vaak ontbreken. Verder zijn er verschillende lessen die zij hebben geleerd voor de opleiding software engineering:

- Training: Voordat een student aan de capstone course kan beginnen moet deze voldoende training krijgen over agile methodes in voorgaande opleidingsonderdelen. Verder moeten studenten, leerkrachten en de faculteit verplicht training krijgen omtrent teamwork, people skills en virtuele teams.
- Collaborative tools: Een breed gamma aan collaboratieve tools moet voorhanden zijn voor de studenten (zoals: WebEx, Skype..).
- Workflow tools: De beste agile methodes zijn hybride, dus er moet gebruik gemaakt worden van tools die hybride agile methodes ondersteunen.
- Developmental tools: Naast het voorzien van tools moeten er ook trainingen gegeven worden voor deze tools te kunnen gebruiken.
- Aesthetic design: Er moet meer nadruk op het esthetisch design gelegd worden aangezien een slecht esthetisch design zorgt voor triljoenen dollars verlies per jaar. De focus mag dus niet alleen liggen op software engineering documentatie.
- Buy vs. Build: Aan software engineering studenten moet als vuistregel geleerd worden dat ze eerst kopen alvorens ze iets bouwen.

Meer informatie over de uitwerking van deze studie kan teruggevonden worden in Rico en Sayani (2009). Verder zullen er verschillende studies met de bijhorende bevindingen over SCRUM in het hoger onderwijs besproken worden.

2.7.1 SCRUM in het hoger onderwijs

Naast het aanleren van de theorie achter Scrum in het hoger onderwijs, zijn er studies die het gebruik van Scrum bij studenten aantonen. De studies zullen in sectie 2.7.1.1 kort worden toegelicht en de resultaten zullen worden besproken in 2.7.7.2.

2.7.1.1 de studies kort toegelicht

In een studie van May et al. (2016) werd een klasoefening gedaan, genaamd "The Ball Game", met de studenten. Het primaire doel van deze oefening is om de studenten zelf de effecten van zelf-organiserende teams te laten ervaren. Zo kunnen de studenten zelf het verschil met de traditionele aanpak ontdekken en de effecten van het Scrum framework begrijpen.

In een tweede studie, een studie van Valle & O'Mara (2015), worden de fundamentele principes van Scrum uitgelegd aan twee verschillende groepen van studenten en moesten de studenten deze ook toepassen in een oefening. Dit was een studie die niet werd afgenomen in een opleidingsonderdeel rond softwareontwikkeling, maar rond projectmanagement. Hier werd er gewerkt met het simuleren van drie opeenvolgende werkdagen met groepen van 3 studenten (1 ScrumMaster en 2 programmeurs). Deze studenten hadden geen programmeerkennis nodig aangezien ze werkten met lijsten die ze moesten invullen om het schrijven van code te simuleren.

Een derde studie, een studie van Landry & McDaniel (2016), had als doel om een set van agile concepten en praktijken samen te stellen die gepast zijn voor een informatiesysteem projectmanagement vak. Het was niet de bedoeling om heel dit vak te herontwerpen, maar eerder om agile aan te leren naast het aanleren van de traditionele projectmanagementmethodes.

Een vierde studie werd gedaan door Mahnic (2010) waarin een opleidingsonderdeel toegewijd was aan het aanleren van Scrum aan studenten. Het eerste doel van het opleidingsonderdeel was het aanleren van Scrum in een 'real world' omgeving waarin ze de methode strikt volgen. Het tweede doel was om de percepties van de studenten over Scrum te ontdekken. Tot slot wilden ze de prestaties van het op Scrum gebaseerde ontwikkelingsproces meten met behulp van het metriekmodel dat in een andere studie van Mahnic wordt voorgesteld.

In een vijfde studie, die werd gedaan door Jiménez en Cliburn (2016), hebben drie verschillende Computer Science opleidingsonderdelen aan dezelfde universiteit gebruik gemaakt van Scrum. De drie opleidingsonderdelen waren de volgende: Software Engineering, Human-Computer Interface Design en Computer Game Technologie.

2.7.1.2 De belangrijkste bevindingen van de studies

Studie 1: "The ball game"

De reacties waren over het algemeen zeer positief. Hierna volgen twee reacties van studenten:

- "Ik vond het leuk hoe zoiets leuks en uitdagend eindigde met een les die ik nooit zal vergeten... initiële schattingen zijn altijd fout... dus doe je best maar overdrijf niet."
- "Ik vond het cool dat verschillende teams doorheen de oefening veranderden in goed functionerende teams en ik wens dat mijn vorige collega's hetzelfde succes zouden kennen."

De implementatie van deze ball game heeft ervoor gezorgd dat enkele studenten de waarde van Scrum en het competitieve voordeel ervan volledig begrepen, waardoor ze een ScrumMaster certificaat behaald hebben. Deze studie toont aan dat er interactieve manieren zijn om de fundamentele principes van Scrum aan te leren aan studenten.

Studie 2: Toepassing van Scrum in een oefening

Na de oefening werd een vragenlijst met vier vragen afgenomen bij deze studenten. Deze vier vragen waren allemaal in de vorm van een likertschaal (met 1=sterk mee oneens tot 5=sterk mee eens). De vragen zullen in tabel 2 geformuleerd worden gevolgd door het gemiddelde en de bijhorende standaarddeviatie van de twee verschillende groepen:

Vragen	μ_1	σ_1	μ_2	σ_2
1. Ik wist weinig over agile methodes voor de oefening	4.3	1.19	3.9	1.60
2. Na de oefening verstond ik de fundamentele concepten en voordelen van agile methodes	4.1	0.70	3.6	0.48
3. De klasoefening was zeer effectief in het tonen van agile (SCRUM)	4.6	0.49	3.9	0.74
4. De klasoefening was zeer innovatief	4.5	0.67	3.9	0.80

Tabel 2: Vragen die werden gesteld in de studie van Valle & O'Mara (2015)

Hieruit kan er worden geconstateerd dat deze oefening voor positieve resultaten zorgde. Verder werd er geconcludeerd dat studenten voorbereid moeten worden voor de wereld die ze gaan betreden na hun studies. Werkgevers zijn consistent in hun eisen dat werknemers innovatief en in staat moeten zijn om succesvol te werken in complexe en veranderende omgevingen. Er moet dus naast het aanleren van de voorspellende (traditionele) methodes ook ruimte zijn voor het aanleren van de adaptieve (agile) methodes. Hierdoor leren studenten nuttige vaardigheden aan die ervoor zullen zorgen dat ze succes zullen boeken in hun werklevens (Valle & O'Mara, 2015).

Studie 3: Agile aanleren naast de traditionele projectmanagementmethodes

De auteurs wilden agile niet behandelen als de "zilvere kogel" en erkende dat niet alle situaties volledig gepast zijn voor de agile aanpak. Er werd gebruik gemaakt van de verschillende onderdelen van PMBOK als een inhoudskader zodat agile op een uitgebreidere manier benaderd werd. Vervolgens lag de focus niet op agile software development, maar eerder op agile projectmanagement.

De resultaten werden bekomen door een RAT (Readiness Assurance Test). Dit is een soort quiz waar studenten eerst individueel (iRAT) tien meerkeuzevragen, die elk op vier punten gingen, moesten beantwoorden. Dezelfde quiz werd achteraf ook gedaan maar dan bij de teams (tRAT). De resultaten wezen erop dat:

- De studenten het agile projectmanagement materiaal even goed onder de knie kregen als de andere inhoud van het vak;
- Het opleidingsonderdeel te zwaar werd door het toevoegen van agile topics, waardoor de midterms resultaten onder het historisch gemiddelde lagen, en;
- De studenten geen accurate schattingen konden maken over de tijd die er nodig was voor de programmeer taken aangezien de focus van de onderwerpen lag op projectmanagement en niet op softwareontwikkeling.

De auteurs halen aan dat ze zich in de toekomst voornamelijk gaan focussen op de burndown charts en Kanban oefeningen of ze willen de 'planning poker' tweakken om dit laatste probleem te

voorkomen. De aanbevelingen die zij maken voor het gebruik van agile projectmanagement zijn de volgende (Landry & McDaniel, 2016):

1. Kies op zijn minst één leerdoelstelling gerelateerd aan agile projectmanagement waarop je de focus legt.
2. Leg de nadruk van de agile projectmanagement concepten op de verschillende PMBOK gebieden (integratie, scope, tijd, kosten, kwaliteit, HR, communicatie, risico, procurement en stakeholder).
3. Aangeraden is om Agile PM te beoordelen, zodat de beheersing van het materiaal door de student onafhankelijk van de rest van de inhoud geïsoleerd kan worden en de tussenkomst kan worden geëvalueerd.
4. Aangeraden is aan om materiaal te reviewen alvorens uitgebreide tests uit te voeren.
5. Aangeraden is om aanvullende literatuurstudies door te nemen voor het ontdekken van concepten, methoden en benaderingen voor het lesgeven van agile PM die mogelijk informatief zijn, maar buiten het bereik van deze studie vallen.

Studie 4: Scrum aanleren in een opleidingsonderdeel

De studenten konden kiezen tussen twee projecten, waarvan het tweede project in samenwerking was met een bedrijf in Slovenië. Bij het eerste project was de leerkracht de klant, de ScrumMaster maar ook de Product Owner. Bij het andere project speelde de vertegenwoordiger van het bedrijf in Slovenië al deze rollen. Ze werkten in totaal met 2 sprints van 30 kalenderdagen en hadden besloten om te werken met twee Daily Scrum vergaderingen per week (één vastgelegd moment waarbij de klant/ScrumMaster/Product Owner aanwezig was en een ander moment waarbij de teams zelfstandig op donderdag moesten samenkomen). Dit omdat er niet verwacht kon worden van een student om elke dag aan het project te werken door andere opleidingsonderdelen. Na elke vergadering diende een bijgewerkt Sprint Backlog document naar de leerkracht doorgestuurd te worden.

Om het perspectief van de student te bekomen zijn er twee verschillende vragenlijsten afgenomen bij de studenten, waarvan de vragen bestonden uit een vijf-punten likertschaal. Het eerste type vragenlijst werd afgenomen na elke sprint om de ervaringen van de studenten van elke iteratie apart te kunnen evalueren. De tweede vragenlijst werd afgenomen aan het einde van het semester met als doel het evalueren van het vak als een geheel en om de meningen van de studenten te verkrijgen over de voordelen van Scrum die er in de literatuur aangehaald worden. Verder werden de sterktes en zwaktes van Scrum tijdens de sprint retrospectives besproken.

De resultaten van de eerste vragenlijst waren positief. Het samenwerken met de ScrumMaster kreeg de hoogste score wat betekent dat het vak goed voorbereid was en toont aan dat goede coaching nodig is wanneer agile methodes geïntroduceerd worden aan beginners. De rol van de ScrumMaster/Product owner zijn cruciaal voor het succes van zo een opleidingsonderdeel. Dit betekent dat de docenten niet enkel de vorderingen van de studenten moeten monitoren, maar ook moeten samenwerken met de studenten waardoor er dus ook meer inspanning van de docenten moeten komen. De studenten waren ook zeer tevreden over de samenwerking met de teamleden. Dit toont aan dat Scrum bijdraagt aan succesvol teamwerk. Voor de resultaten van de andere vragen kan je kijken naar de tabel die je kan terugvinden in bijlage 1. In de finale vragenlijst vonden 86.67%

van de studenten het opleidingsonderdeel nuttig en interessant. Verder waren studenten het er sterk mee eens dat Scrum hen helpt om succes te delen doorheen het project, helpt om het product op te delen in beheersbare brokken, alles zichtbaar maakt voor iedereen, klantenrelaties verbetert, een cultuur creëert waar iedereen verwacht dat het project zal slagen, en klanten in staat stelt om op tijd product incrementen te zien. Ook was er genoeg bewijs dat Scrum zorgt voor betere communicatie en klantenfeedback over hoe het product echt moet werken. Scrum zorgde er ook voor dat de studenten op tijd begonnen in plaats van alles uit te stellen. Verder heeft de docent gebruik gemaakt van verschillende meetinstrumenten zoals burndown charts, EVM (earned value method) en kostenprestatieindexen (SPI en CPI) (deze meetinstrumenten kan je terugvinden in het ander werk van Mahnic). Zo kon er een volledige analyse van de projectprestatie gedaan worden.

Tot slot is er vastgesteld dat studenten liever praktische lessen krijgen waarbij ze iets aanleren in plaats van theoretische hoorcolleges. Uit ervaring is er ook vastgesteld dat er een goede balans moet zijn tussen coaching en zelforganisatie wanneer Scrum aan beginners wordt aangeleerd. Zelforganisatie is nuttig op vlak van ervaringsgericht leren door ontdekking en participatie, terwijl coaching belangrijk is voor het bereiken van een kwalitatief eindproduct.

Studie 5: toepassing van Scrum in drie verschillende opleidingsonderdelen

De resultaten van deze studie waren ook positief. Ook hier zijn er studenten die bepaalde opmerkingen aanhaalden:

- Studenten antwoordden dat Scrum efficiënt is en dat het hen aanmoedigt om doelen op te stellen voor elke sprint en deze te voltooien.
- Een andere student vermeldde dat hij de principes van Scrum nu ook toepast in andere opleidingsonderdelen.

Twee negatieve opmerkingen die werden aangehaald zijn de volgende:

- Een student meende dat Scrum beter gebruikt kan worden in een professionele omgeving in plaats van een schoolomgeving omdat het volgens hem niet effectief gebruikt wordt door de mensen.
- Een andere student melde dat hun team gedurende een standup meeting niks te rapporteren had door het vele werk dat ze hadden voor andere opleidingsonderdelen en werd er aangehaald dat motivatie ontbrak of dat er meer ervaring nodig was bij sommigen studenten.

Deze studie laat zien dat het toepassen van Scrum zeker positief is, maar dat de hoger onderwijscontext het niet helemaal toelaat om Scrum volledig en effectief toe te passen.

2.7.2 XP in het hoger onderwijs

In een studie van Melnik en Maurer (2005), werd het gebruik van agile methodes geïmplementeerd in software engineering opleidingsonderdelen aan de Universiteit van Calgary en Southern Alberta Institute of Technology. Deze studie heeft drie jaar lang geduurd en had drie doelen, namelijk: 1) om de percepties van agile praktijken vanuit het studentenperspectief te verkennen; 2) om verbanden te onderzoeken tussen studentenpercepties van agile methoden in het algemeen en hun

percepties van individuele praktijken; en 3) te onderzoeken hoe de percepties van studenten variëren afhankelijk van het academische programma, de leeftijd en de voorkennis.

De studie richt zich op agile engineering-praktijken die voortkomen uit extreme programming. Het instrument van de studie was een vragenlijst die betrekking heeft tot de drie onderzoeksdoelstellingen en deze werd ingevuld door 240 studenten. Uit deze vragenlijst bleek dat 78% gelooft dat XP de productiviteit van kleine teams vergroot, 76% vond dat XP de kwaliteit van code verbetert en 65% zou XP aanraden aan een bedrijf waar ze werken of in de toekomst zouden gaan werken. Verder is er uit de resultaten geen significant verschil ondervonden tussen de verschillende niveaus van opleidingen. In het algemeen tonen de resultaten aan dat een groot deel van de studenten agile praktijken accepteren en liefhebben. Dit geldt voor alle leeftijden en verschillende graden van voorkennis. Tot slot zijn er kwalitatieve inzichten dat het werken als een agile team de ontwikkeling van professionele vaardigheden bevordert (communicatie, toewijding, samenwerking en aanpassingsvermogen)

2.7.3 Algemene conclusie studies

Deze studies tonen aan dat er wel al enige moeite is gestoken om Scrum of XP te implementeren in het hoger onderwijs. Over het algemeen waren de bevindingen zeer positief en laat dit zien dat Scrum (en XP in het geval van een software ontwikkelingsopleidingsonderdeel) op een bepaalde manier implementeerbaar is in het hoger onderwijs mits de studenten en de docenten zich hiernaar aanpassen. De bevindingen die bekomen zijn uit deze studies zullen gebruikt worden om enkele vragen op te stellen in de enquêtes die zullen volgen in hoofdstuk 3. De vragen die worden opgesteld zijn bedoeld om deze resultaten te bevestigen, eventueel nieuwe bevindingen te bekomen en om te kijken welke principes van agile toepasbaar zijn in het hoger onderwijs.

2.7.4 Agile Learning

Naast het toepassen van de agile methodes in het hoger onderwijs, is er ook een ander concept aan het licht gekomen, namelijk agile learning. Agile learning past de processen en de principes van agile softwareontwikkeling toe in de context van leren. De auteur heeft de vier kernwaardes van het agile manifesto aangepast zodat deze toepasbaar zijn voor agile learning. De vier kernwaardes worden hierna beschreven samen met de toepassing in het onderwijs (Lang, 2017):

1. Individueen en hun onderlinge interacties boven processen en hulpmiddelen: Het suggereert dat de docent zich **richt op het één-op-één werken met studenten** en **flexibel is** in het aanpassen van de processen en hulpmiddelen die in het klaslokaal worden gebruikt
2. Werkende software boven allesomvattende documentatie: Het suggereert dat de **focus** verschuift van studenten die rapporten schrijven naar **studenten die iets produceren** dat kan worden gebruikt in een professionele omgeving
3. Samenwerking met de klant boven contractonderhandelingen: Het suggereert de docent om **samen te werken met studenten** in plaats van strikte handhaving van opdrachten en bijbehorende regels
4. Inspelen op verandering boven het volgen van een plan: Dit benadrukt verder de noodzaak voor de docent bereid te zijn om af te wijken van het traditionele semester cursusschema en

in plaats daarvan het **schema aan te passen naar de behoeften van studenten** wanneer ze zich voordoen

Het doel van de agile learning kernwaarde is het verbeteren van het vermogen van de docent om het leren in een agile leerervaring te vergemakkelijken. Hier wordt benadrukt dat de instructeur zich zal moeten aanpassen zodat de lessen op een agile manier gegeven kunnen worden.

Lang (2017) heeft onderzoek gedaan in een computer informatiesysteem opleidingsonderdeel over webontwikkeling. Dit opleidingsonderdeel was compleet herontworpen om agile learning een semester lang te implementeren. Na dit semester werd een vragenlijst gegeven aan de studenten, waarvan 34 studenten deze hadden ingevuld. Na het analyseren van de resultaten waren er twee voordelen en twee nadelen ontdekt.

De voordelen:

- Het combineert leren met het toepassen van leren. Zo wordt de tijdsvertraging hiertussen verkleind. De studenten leren iets en passen het ook meteen toe.
- Het laat toe om meer en sneller te falen. Door te werken met meerdere sprints, kunnen de studenten hun tekortkomingen van het begrijpen van bepaalde materie vaker en sneller detecteren.

De nadelen:

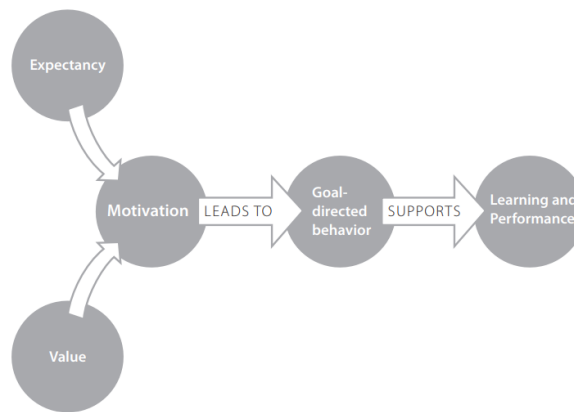
- Het duurt langer dan traditioneel projectgebaseerd leren.
- Het is makkelijker voor studenten om achterstand op te lopen dan bij traditioneel projectmatig leren. Dit kan voor problemen zorgen wanneer een student een onderdeel niet begrijpt aangezien de leerstof op elkaar verder bouwt.

Verder was er een grote voorkeur van de studenten om agile learning ook toe te passen in opleidingsonderdelen waar het traditionele projectgebaseerde leren toegepast wordt. Dit laat zien dat studenten bereid zijn om agile learning te leren en toe te passen. Hieruit kan afgeleid worden dat zowel de houding van studenten als de houding van de docenten aangepast moeten worden om agile learning mogelijk te maken.

2.8 Motivatie van studenten

Aangezien in deze literatuurstudie het onderwerp 'motivatie' aan bod is gekomen en omdat motivatie in het verdere verloop van deze thesis nog aan bod zal komen, is het belangrijk om na te gaan wat ervoor zorgt dat studenten gemotiveerd zijn om te leren. In een hoofdstuk van een boek van Ambrose, Bridges, DiPietro, Lovett en Norman (2010) werd er onderzocht welke elementen een grote invloed hebben op motivatie bij studenten.

Er zijn twee concepten die centraal staan om motivatie te begrijpen. Ten eerste heb je de waarde van een doel en ten tweede heb je de verwachtingen voor het succesvol bereiken van dat doel. Dit wordt voorgesteld in figuur 9. Deze figuur illustreert de wisselwerking tussen waarde en verwachtingen om de mate van motivatie om deel te nemen aan doelgericht gedrag te beïnvloeden. Deze concepten worden in de volgende paragrafen kort toegelicht.



Figuur 9: Impact van waarde en verwachting op leren en prestatie. (Ambrose et al., 2010)

2.8.1 Doelen

Er zijn verschillende doelen die studenten kunnen hebben die hun gedrag bepalen. Zo bestaan er prestatiedoelen, leerdoelen, werk-vermijdende doelen, affectieve doelen en sociale doelen. Ten eerste zijn er de **prestatiedoelen**. Deze omvatten het beschermen van een gewenst zelfbeeld en het projecteren van een positieve reputatie en publieke persoonlijkheid. Dit wil zeggen dat studenten die prestatiedoelen hebben, zich bezighouden met normatieve normen en proberen te doen wat nodig is competent en intelligent te lijken, maar ook om status, erkenning en waardering te krijgen. Ten tweede heb je de **leerdoelen**. Studenten proberen competentie te verwerven en echt leren wat een activiteit of taak hen kan leren. Deze studenten zullen bijvoorbeeld de leerstof beter en dieper begrijpen, maar zullen ook hulp zoeken wanneer dat nodig is, volhouden wanneer er zich moeilijkheden voordoen en zullen zich comfortabel voelen met uitdagende taken. Vervolgens heb je de **werkvermijdende doelen**. Studenten met dit doel voor ogen hebben meestal de neiging om het werk zo snel mogelijk af te handelen met zo weinig mogelijk inspanning. Deze studenten zijn niet gedreven om iets bij te leren en lijken vaak vervreemd, ontmoedigd of ontkoppeld. Wat hier wel moet aangehaald worden is dat dit vaak context-specifiek is. Een student kan zich inzetten voor één opleidingsonderdeel, maar vermijdt het werk in een ander opleidingsonderdeel.

Wanneer een activiteit meer dan één doel doet voltooien, is het vaak zo dat de motivatie om deze activiteit uit te voeren groter is. Hierbij kan aangehaald worden dat een student die een opleidingsonderdeel volgt en bijvoorbeeld een leerdoel (iets grondig leren) heeft, maar daarnaast ook een affectief doel (een stimulerende activiteit) en een sociaal doel (vrienden maken) heeft, deze student meerdere doelen tegelijkertijd kan voltooien waardoor de motivatie kan toenemen (Ambrose et al., 2010).

2.8.2 Waarde

Er zijn drie brede determinanten van subjectieve waardering voor prestatieafhankelijke activiteiten en doelen. De eerste is de **eindwaarde**, die de voldoening vertegenwoordigt die men krijgt door het beheersen en het voltooien van een doel of taak. De tweede bron van waarde is de **intrinsieke waarde**. Deze vertegenwoordigt de voldoening die wordt behaald door het uitvoeren van de taak. Deze waarde is nauw verbonden met de specifieke inhoud van een taak en wordt ook wel intrinsieke

motivatie genoemd. Tot slot is er de **instrumentele waarde**. Het vertegenwoordigt de mate waarin een activiteit of doel helpt om andere belangrijke doelen te bereiken, zoals het verkrijgen van wat traditioneel wordt aangeduid als extrinsieke beloningen (extrinsieke motivatie). Bijvoorbeeld, een student die studeert en verwacht dat hij of zij later een groot loon en status zal krijgen door de studie die hij of zij heeft gedaan, is gemotiveerd om te studeren en de lessen bij te wonen op basis van de instrumentele waarde die de lessen bieden voor het gewenste salaris en status (Ambrose et al., 2010).

Één van deze bronnen kan leiden tot motivatie, maar in de realiteit kunnen deze drie bronnen in combinatie voorkomen en elkaar aanvullen. Het is van belang dat deze waardes voorkomen bij studenten zodat ze gemotiveerd worden om te leren.

2.8.3 Verwachtingen

Waarde alleen is niet voldoende om gedrag te motiveren. Mensen zijn ook gemotiveerd om doelen en resultaten na te streven waarvan zij denken dat ze het met succes kunnen bereiken. Anderzijds, in het geval dat ze de doelen en resultaten denken niet te kunnen bereiken, is de kans groter dat de motivatie minder is. Verder zullen twee vormen van verwachtingen worden besproken.

Ten eerste zijn er de **uitkomstverwachtingen**. Deze weerspiegelen de overtuiging dat specifieke acties een gewenste uitkomst zullen opleveren. Zo zijn er positieve en negatieve uitkomstverwachtingen. Een voorbeeld van een positieve uitkomstverwachting is wanneer een student denkt dat als hij alle leerstof leest en goed participeert in de discussies die gevoerd worden in de lessen, hij het leermateriaal genoeg onder de knie krijgt om op het examen te kunnen slagen. Om ervoor te zorgen dat studenten gemotiveerd zijn om deel te nemen aan het gedrag dat resulteert in leren, moeten ze geloven dat er een verband bestaat tussen dat gedrag en de resultaten die ze verlangen.

Ten tweede zijn er de **waakzaamheidsverwachtingen**. Deze vertegenwoordigen de overtuiging dat men in staat is een werkwijze te identificeren, organiseren, initiëren en uit te voeren die een gewenste uitkomst zal opleveren. Een student moet dus, naast het slagen op een test, geloven dat hij of zij bekwaam is om het werk te doen dat nodig is om te kunnen slagen. Wanneer studenten een doel succesvol bereiken en dit doel is bereikt door interne (bijvoorbeeld eigen talenten en vaardigheden) of controleerbare (bijvoorbeeld eigen moeite of doorzettingsvermogen) oorzaken, de kans groot is dat hij of zij verwacht dat het resultaat van een volgend doel of opdracht positief zal zijn. Als de oorzaak van succesvol slagen ligt bij externe (bijvoorbeeld een makkelijke opdracht) of oncontroleerbare (bijvoorbeeld geluk) oorzaak, verwacht hij of zij minder succes.

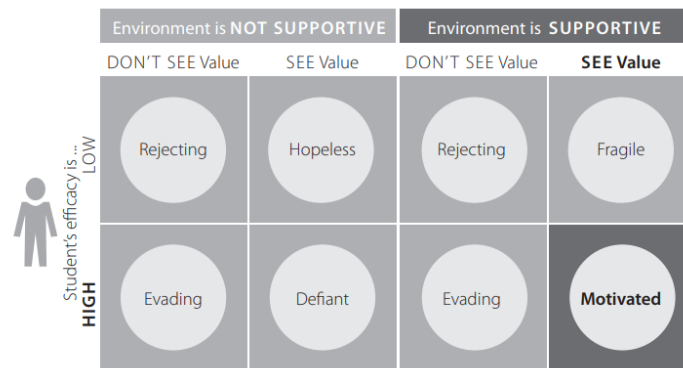
Tot slot, wanneer een student faalt kan hij of zij de draad weer oppikken (en gemotiveerd blijven of worden) door de oorzaak te vinden en zich hiernaar aan te passen (Ambrose et al., 2010).

2.8.4 Omgeving

Waarde en verwachtingen worden beïnvloed door de omgeving. De omgeving kan ondersteunend of niet ondersteunend zijn. Een voorbeeld van ondersteunend is wanneer je klasgenoten er voor je zijn en je willen helpen wanneer je met een probleem zit, maar ook wanneer de docent benaderbaar is. Deze twee voorbeelden zouden ervoor kunnen zorgen dat motivatie zal stijgen.

2.8.5 De interactie tussen drie elementen

Er zijn drie hefboomen die hiervoor besproken werden en die invloed hebben op motivatie, namelijk waarde, werkzaamheidsverwachtingen en ondersteunend omgeving. Als er één van deze hefboomen verwaarloosd wordt, zal motivatie hier sterk onder lijden. In figuur 10 zijn de verschillende gedragingen te zien die het resultaat zijn van de interactie tussen de drie hefboomen.



Figuur 10: Interactief effect van de omgeving (Ambrose et al., 2010).

Zo zal een student enkel volledig gemotiveerd zijn wanneer de werkzaamheidsverwachtingen hoog liggen, wat wil zeggen dat ze vertrouwen hebben in hun kunnen, de waarde gezien wordt in een taak en de student ondersteund wordt door de omgeving. Deze studenten proberen nieuwe kennis te leren, te integreren en toe te passen en leersituaties te zien als kansen om hun begrip uit te breiden (Ambrose et al., 2010).

2.8.5 Strategieën om waarde en positieve verwachtingen te ontwikkelen

In deze sectie zal in het kort een opsomming gegeven worden over de strategieën om waarde en positieve verwachtingen te creëren (Ambrose et al., 2010).

Strategieën om waarde te creëren bij de studenten en dus het belang van een opleidingsonderdeel te doen inzien:

- Verbind de leerstof met de interesses van de studenten. Dit kan gedaan worden door hen concreet te wijzen op de bruikbaarheid van de kennis en vaardigheden die ze vergaren in het opleidingsonderdeel in het verdere verloop van hun (beroeps)leven.
- Speel in op de leefwereld van de studenten door te werken met voorbeelden uit de dagelijkse realiteit van de studenten en geef hen taken die gebaseerd zijn op een real-world scenario. Op deze manier zal de motivatie stijgen.
- Maak verbindingen tussen het gegeven opleidingsonderdeel en andere opleidingsonderdelen die volgen, om de studenten beter te doen begrijpen wat de waarde van elk opleidingsonderdeel is.
- Toon de relevantie van het hoger niveau van vaardigheden voor het toekomstige professionele leven van de student.
- Identificeer en beloon wat je waardeert. Hiermee wordt bedoeld dat je aan de studenten moet communiceren wat je verwacht van hen en als ze aan deze criteria voldoen, hier ook voor 'beloond' worden door hen te beoordelen op basis van deze criteria.

- Toon je eigen passie en enthousiasme voor het opleidingsonderdeel. Dit gedrag kan aanstekelijk werken en kan de nieuwsgierigheid van de studenten opwekken.

Strategieën die studenten helpen om positieve verwachtingen te ontwikkelen

- Zorgen voor afstemming van doelstellingen, beoordelingen en instructiestrategieën. Op deze manier wordt er het leren ondersteund en zullen de studenten zich meer bekwaam voelen om doelen te bereiken.
- Identificeer het gepaste uitdagingsniveau voor je studenten en stem de taken hierop af. De studenten moeten uitdagende taken of doelen krijgen, maar de doelen moeten haalbaar zijn. De voorkennis van je studenten in kaart brengen is hierbij een mogelijkheid. Dat kan gedaan worden door zich te informeren bij de docenten van een voorafgaand opleidingsonderdeel
- Voorzie vroegtijdig succes voor de studenten. Wanneer de studenten vroegtijdig in het semester succes boeken, groeit hun vertrouwen en worden ze meer gemotiveerd.
- Vertaal je verwachtingen duidelijk naar de studenten en vertel de studenten wat je verwacht dat ze doen om de doelen te bereiken, dus door concreet te maken wat voor gedrag er van hen verwacht wordt.
- Geef studenten voldoende constructieve feedback. Aangezien feedback ervoor zorgt dat studenten geïnformeerd worden over de vooruitgang die ze boeken richting een doel, kan het een krachtige manier zijn om de studenten te motiveren.
- Wees eerlijk ten opzichte van de studenten. Zorg ervoor dat de standaarden en criteria die gebruikt worden om het werk van studenten te beoordelen eerlijk worden gebruikt. Als studenten denken dat hun werk anders werd beoordeeld dan het werk van een medestudent, zal de verwachting voor succes dalen.
- Leer de studenten om de juiste elementen toe te schrijven aan succes en mislukking. Zorg er bijvoorbeeld voor dat wanneer een student niet goed is met cijfers, hem of haar te wijzen op dat de oorzaak van een slecht punt niet hieraan ligt. Beschrijf de juiste studie strategie of studievaardigheden die ze kunnen gebruiken om een bepaald doel te bereiken. Hierdoor passen ze hun gedrag aan en verhogen ze hun slaagkansen.

2.8 Conclusie literatuurstudie

Om de basis van projectmanagement uit te leggen, is in sectie 2.2 aangehaald wat een methodologie is en wat projectmanagement inhoudt. Verder werden in sectie 2.3 de traditionele projectmanagement methodes aangehaald om een beter beeld te bekomen wat projectmanagement werkelijk is. In sectie 2.4 wordt agile softwareontwikkeling uitgebreid uitgelegd alsook de twee methodes XP en Scrum om de agile concepten duidelijk voor te stellen. Vervolgens wordt in sectie 2.5 een meer algemeen beeld gegeven over agile projectmanagement aangezien de focus van deze thesis niet ligt op het behandelen van specifieke methodes. In deze sectie werden ook de risico's aangehaald op vlak van projectmanagement die bij agile methodes mogelijk zijn.

Het is zo dat agile projectmanagement al een lange tijd wordt gebruikt in de bedrijfswereld en wordt elke dag meer en meer gebruikt in kleine en middelgrote bedrijven. Het zorgt voor betere productiviteit en voegt meer waarde toe wanneer er gewerkt wordt met kleine cross-functionele en

gemotiveerde teams waarbij samenwerking en goede communicatie centraal staan. Echter kan er verder onderzoek gedaan worden naar het toepassen van APM in grote bedrijven, maar dit zal verder niet behandeld worden in deze thesis.

Verder werden in sectie 2.6 de verschillen tussen de traditionele projectmanagementmethodologie en de agile projectmanagementmethodologie vergeleken met elkaar om aan te tonen hoe ze verschillen van elkaar op vlak van het managementaspect. Het was hier duidelijk dat ze op veel vlakken verschillen van elkaar.

In sectie 2.7 zijn verschillende studies aangehaald die zowel de toepassing van Scrum en Xp in het hoger onderwijs onderzochten. Hieruit waren voornamelijk positieve resultaten voortgekomen en laat zien dat het wel degelijk een introductie in het hoger onderwijs kent. Deze studies werden voornamelijk gedaan opleidingsonderdelen rond softwareontwikkeling. Hier leren de studenten op een agile manier software schrijven en hoe ze op een goede manier kunnen communiceren en samenwerken. Dit zorgt voor betere software en dit bereid de studenten voor om de agile bedrijfswereld in te stappen. Onderzoek naar het toepassen van agile projectmanagement bij projecten in andere opleidingsonderdelen van een ander soort is nog bijna niet gebeurd, met enkele uitzonderingen zoals bij de studie van Valle & O'Mara (2015).

Tot slot werd in sectie 2.8 uitgelegd wat ervoor zorgt dat studenten gemotiveerd zijn of kunnen worden. Motivatie speelt een grote rol bij het implementeren van een nieuwe manier van werken, namelijk agile projectmanagement, in het hoger onderwijs. Er zal doorheen deze thesis nog verschillende keren verwezen worden naar deze sectie aangezien de docenten alsook de studenten zich moeten aanpassen en er dus voor moeten zorgen dat ze gemotiveerd zijn om agile projectmanagement toe te kunnen passen in het hoger onderwijs.

Uit deze literatuurstudie kan er worden afgeleid dat agile projectmanagement toepasbaar is in het hoger onderwijs mits enkele wijzigingen worden doorgevoerd. Wat er moet veranderen aan de houding van studenten, docenten en het onderwijssysteem moet worden onderzocht. Dit zal in het verdere verloop van deze thesis aan bod komen.

Hoofdstuk 3: Agile projectmanagement in het hoger onderwijs

3.1 Inleiding

Agile projectmanagement wordt steeds meer en meer gebruikt in het hoger onderwijs zoals blijkt uit de studies die zijn gedaan en werden aangehaald in de literatuurstudie. Universiteiten zoals de Universiteit Hasselt proberen agile methodes toe te passen in specifieke opleidingsonderdelen. Zoals in de literatuur vermeld werd, is het zo dat agile projectmanagement zijn oorsprong kent in softwareontwikkeling. Dit betekent dat dit is ontstaan om softwareontwikkelingsgerelateerde problemen efficiënter en effectiever op te kunnen lossen. Aangezien er een opkomst is van agile projectmanagement in het hoger onderwijs, moet er onderzocht worden of deze methodologie geschikt is voor het toepassen op projecten of groepswerken die in het hoger onderwijs voorkomen. In deze thesis, zoals in de inleiding al vermeld, is het de bedoeling om richtlijnen te formuleren die professoren en studenten kunnen hanteren, om zo agile praktijken ook in andere domeinen dan louter softwareontwikkeling toe te kunnen passen.

In dit hoofdstuk zullen de twaalf leidende principes van agile projectmanagement vertaald worden naar principes die toepasbaar zijn in het hoger onderwijs. Dit zal, zoals in de onderzoeksaanpak aangehaald werd, gedaan worden op basis van het onderzoeken van een case. Om de case te onderzoeken werd er gebruik gemaakt van enquêtes die werden afgenomen bij studenten van de Universiteit Hasselt. Deze enquêtes hadden als doel om de twaalf principes te vertalen naar een hoger onderwijscontext en verder te onderbouwen, maar ook om voorkeuren over bepaalde elementen in projectwerk te achterhalen. Hieruit zullen bepaalde richtlijnen geformuleerd kunnen worden die zullen helpen om het antwoord op de centrale onderzoeksvraag beter te kunnen onderbouwen. In sectie 3.2 zal er informatie gegeven worden over deze enquêtes en de steekproef, zal de voorkennis van agile projectmanagement van de respondenten geschetst worden en zullen de ervaringen van de respondenten over het project, waar zij volgens de agile methode Scrum werken, voorgesteld worden. Vervolgens worden in sectie 3.3 de twaalf verschillende principes voorgesteld met de bijhorende bevindingen uit de enquêtes. Verder zullen in sectie 3.4 de beperkingen van deze studie aangehaald worden. Tot slot zal in 3.5 de conclusie aangehaald worden waarin een antwoord op de eerste deelvraag geformuleerd wordt.

3.2 Een onderzoek naar agile projectmanagement in het hoger onderwijs

In de literatuurstudie zijn de vier kernwaardes van agile projectmanagement aan bod gekomen gevolgd door de twaalf leidende principes van agile. Vervolgens zijn de risico's verbonden aan agile projectmanagement aangehaald en zijn er resultaten van studies omtrent de implementatie van APM in het hoger onderwijs. Aan de hand van deze kernwaardes, principes, risico's en resultaten van de studies, zijn er in dit hoofdstuk vragen opgesteld om te achterhalen welke van de twaalf leidende principes toepasbaar zijn in een hoger onderwijscontext. Er zijn twee enquêtes met deze vragen opgesteld en deze zijn vervolgens afgenomen bij studenten (39 respondenten) van het opleidingsonderdeel "agile business model design" aan de Universiteit Hasselt. Wat hier duidelijk is, is dat het geen random sample betreft maar een theoretische sample. Op deze manier kunnen de resultaten analytisch veralgemeend worden, maar niet statistisch. De reden waarom het niet statistisch veralgemeend kan worden is omdat de voorkeuren van de studenten die deze enquêtes

hebben ingevuld, niet persé de voorkeuren zijn die andere studenten zouden hebben. Hiervoor zouden deze enquêtes bij andere studenten moeten worden afgenomen, zodat de resultaten meer veralgemeend kunnen worden. De groep van 39 studenten bestaat uit twaalf beleidsinformatica studenten waarvan twee studenten vrouwelijk zijn, en 27 mannelijke informaticastudenten. In dit opleidingsonderdeel werken de studenten volgens de principes van Scrum om zo in periodes (of sprints) van twee weken een fictief bedrijf op te starten en een business model voor dit bedrijf uit te werken en voor te stellen. Verder zijn deze twee enquêtes ook ingevuld door twee studenten van de opleiding Industriële wetenschappen en door twee Fysicastudenten die samen aan een project werkten. Hierdoor waren er 43 respondenten in totaal. Ook de omvang van de steekproef speelt hier een rol waardoor de veralgemeenbaarheid beperkt blijft. De gemiddelde leeftijd van al de respondenten ligt tussen de 20 en 22 jaar. Het waren voornamelijk masterstudenten met uitzondering op een paar studenten die nog opleidingsonderdelen volgden uit voorgaande studie jaren op het moment van het onderzoek.

De eerste enquête werd afgenomen in week één van het tweede semester van het academiejaar 2017-2018. Deze eerste enquête was een inleidende enquête om te kijken wat de student al wist over agile projectmanagement en om de voorkeuren voor bepaalde zaken in een project te achterhalen. Dit was dus ook om te kijken of er bepaalde aspecten zijn die tegen de agile methodologie ingaan, maar waar studenten in veel gevallen belang aan hechten, zoals bijvoorbeeld een leider in een groepswerk.

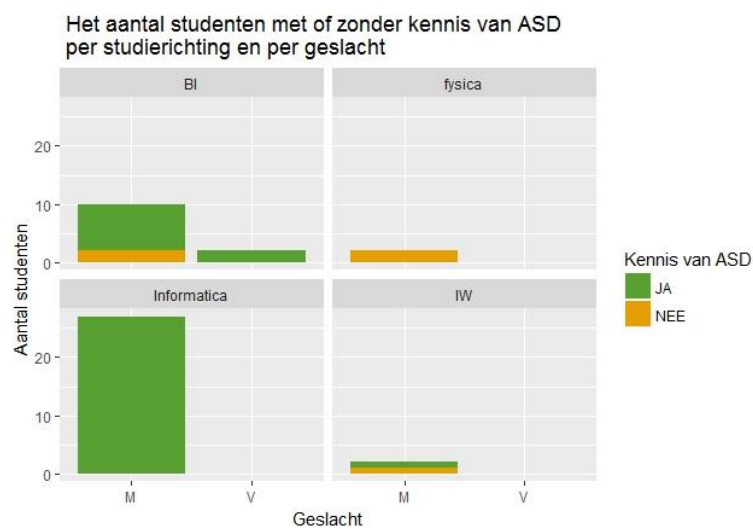
De tweede enquête werd afgenomen in week zeven van het tweede semester van het academiejaar 2017-2018. Deze tweede enquête had als doel om te achterhalen of bepaalde manieren van denken van studenten veranderd waren ten opzichte van de denkwijzen aan de start van het projectwerk. De resultaten dienden ook als aanvulling op de resultaten van de inleidende enquête. Van de 39 respondenten van het opleidingsonderdeel 'agile business model design' hebben 35 studenten deze tweede enquête ingevuld. De reden hiervoor is dat er drie studenten afwezig waren en één student de enquête niet serieus heeft genomen en deze grotendeels open heeft gelaten, waardoor deze uit de analyse is gelaten. Verder hebben de twee studenten van Fysica en de twee studenten van Industriële Wetenschappen de tweede enquête niet ingevuld aangezien deze per mail verzonden is, gevolgd door een reminder met geen respons als gevolg. De tweede enquête werd dus afgenomen bij 35 respondenten.

Eerst zal er kort besproken worden of de studenten bekend waren met agile projectmanagement alvorens ze aan het project begonnen. Daarna zal de ervaring van het toepassen van Scrum gedurende zeven weken besproken worden.

Vervolgens zullen de twaalf principes één voor één besproken worden met de bekomen resultaten uit de enquêtes om zo een antwoord te kunnen formuleren op de eerste deelvraag van deze thesis: "Welke principes van agile projectmanagement zijn toepasbaar in het hoger onderwijs?". Om het onderscheid te maken tussen de inleidende enquête en de tweede enquête worden de grafieken van de inleidende enquête voorgesteld met een groene kleur en deze van de tweede enquête met een blauwe kleur. Sommige onderdelen (zoals de ranking en likert-schaal) hebben in beide enquêtes dezelfde kleur, maar hierbij wordt duidelijk vermeld bij welke enquête de figuur hoort.

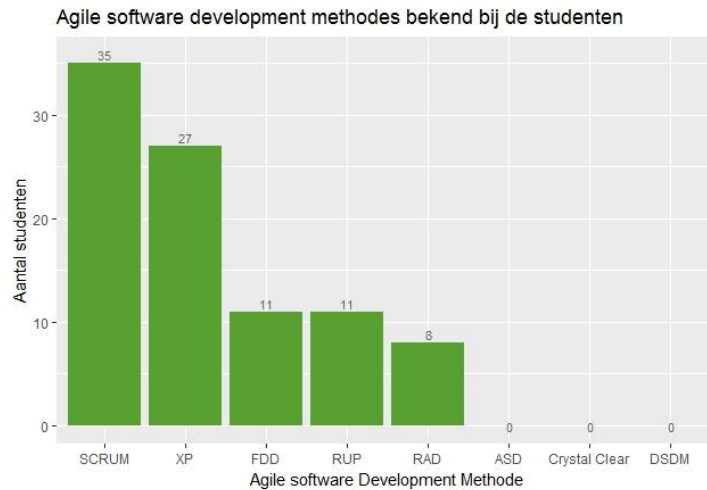
3.2.1 Voorkennis van Agile projectmanagement door studenten

Aan het begin van de inleidende enquête werd er aan de studenten gevraagd of ze al ooit gehoord hadden van ASD (agile software development). Zoals te zien is in figuur 11 kan er worden afgelezen dat bij de opleiding Beleidsinformatica, acht van de tien mannen alsook de twee vrouwen al ooit van ASD gehoord hadden. Van de opleiding informatica hadden alle studenten al gehoord van ASD. De beleidsinformatica- en de informaticastudenten zijn enkele ASD-methodes tegengekomen in bepaalde voorafgaande opleidingsonderdelen. De twee fysica studenten hadden nog nooit gehoord van ASD en één van de twee industriële wetenschappenstudenten had er ooit van gehoord. Dit wil zeggen dat een grote meerderheid, namelijk 38 studenten, al ooit gehoord had van agile softwareontwikkeling.



Figuur 11: Voorkennis ASD door studenten

Aangezien een grote meerderheid voorkennis had van ASD was het interessant om te weten te komen van welke methodes ze al ooit gehoord hadden. In figuur 12 worden de verschillende methodes voorgesteld op de x-as met de bijhorende aantallen op de y-as. Op deze figuur is af te lezen dat 35, respectievelijk 27, van de respondenten al ooit gehoord had van Scrum en XP. Sommigen hadden ook kennis van andere methodes zoals FDD, RUP en RAD, maar geen enkele respondent had al ooit gehoord van adaptive software development, Crystal Clear of DSDM. Verder zijn er door zeven studenten enkele andere ASD-methodes aangehaald die niet in de lijst stonden vermeld. Zo werd door twee studenten kanban aangehaald en vertelde een student dat hij al ooit van pair programming, wat zoals eerder vermeld een onderdeel van XP is, gehoord had.



Figuur 12: De ASD's waarvan de studenten al hebben gehoord

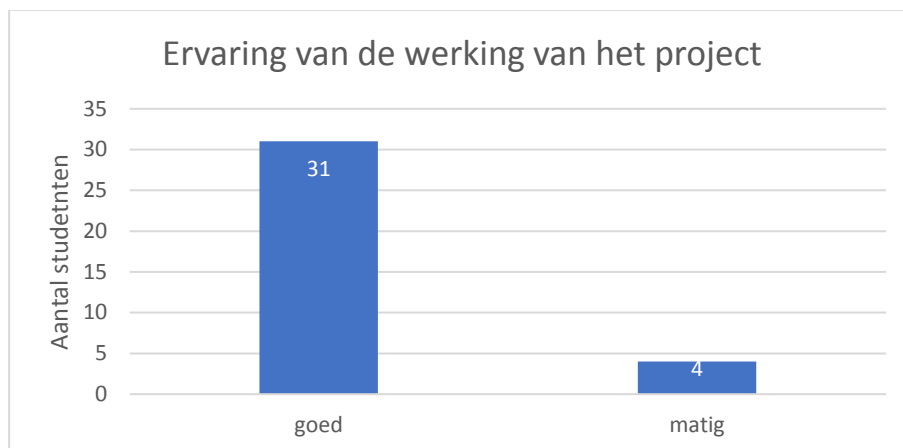
Vervolgens werd aan de studenten gevraagd of ze een goede ervaring hadden met het gebruik van de ASD's die ze kennen (hier werden alleen de 38 studenten die een ASD kennen in acht genomen). 22 studenten vertelden dat ze er een goede ervaring mee hadden en haalden voorbeelden aan van de voordelen die ASD's bieden. Zo werd er aangehaald dat ASD's zorgen voor snellere feedback, incrementele ontwikkeling en het constant betrekken van de klant gedurende het proces. Vervolgens waren er tien studenten die vertelde dat ze er geen goede ervaring mee hadden, maar de uitleg die ze hierbij gaven was dat ze de aangeduide ASD-methode(s) nog nooit hebben toegepast. Tot slot waren er zes studenten die niets hadden ingevuld en hierbij ook geen commentaar gaven.

Ook werd aan de studenten de vraag gesteld of ze anderen zouden aanraden om een agile methode te gebruiken. Hierop hebben 31 van de 38 studenten "ja" geantwoord met als meest voorkomende reden dat veel IT- en algemene bedrijven een ASD-methode of meerdere ASD-methodes gebruiken en dat de voordelen ervan doorwegen ten opzichte van de voordelen van traditionele projectmanagementmethodologieën. Sommigen van de respondenten gaven ook als mening dat het volgens hen zorgt voor een snellere en effectievere manier om software te ontwikkelen. Hier kan al direct een link gelegd worden met de literatuurstudie, want in de literatuurstudie bij de risico's verbonden aan APM werd aangehaald dat er zich een probleem voordoet wanneer de teamleden onvoldoende kennis van de agile methodes hebben. Aangezien de meerderheid kennis heeft van de agile methodes, kan ervan uit gegaan worden dat de problemen beperkt zullen blijven. Er moet wel aangehaald worden dat de interpretaties van bepaalde elementen van een methode door verschillende mensen kunnen verschillen. Dit kan ervoor zorgen dat bepaalde elementen foutief worden geïmplementeerd. Een oplossing hiervoor is een goede training over de agile methodes te geven alvorens het project begint.

3.2.2 Ervaring met werken volgens Scrum binnen projectwerk

Zoals bij sectie 2.1 werd aangehaald is er in week 7 een tweede enquête afgenomen bij de 35 beleidsinformatica- en informaticastudenten van het opleidingsonderdeel 'agile business model design'. De eerste vraag die werd gesteld in deze enquête heeft betrekking op de ervaring van de

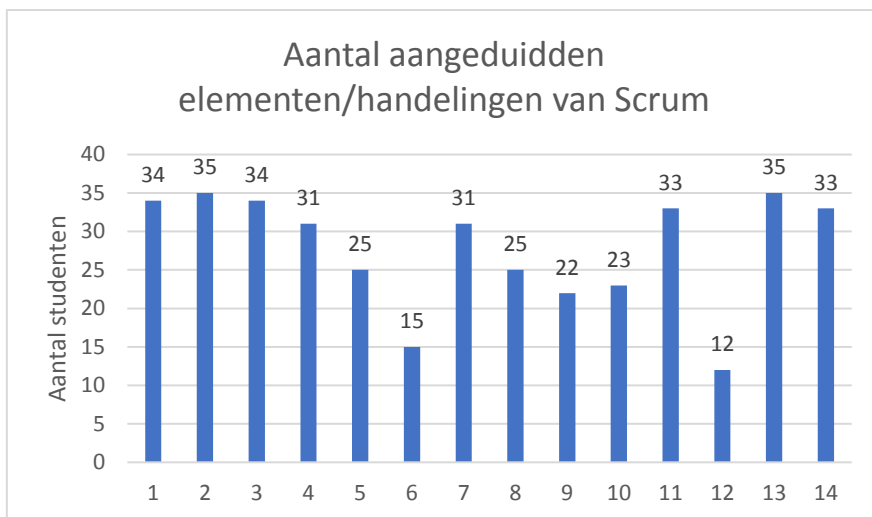
werking van het project en of de studenten goede vooruitgang boekten richting het voltooien van het project.



Figuur 13: De werking van het project

In figuur 13 kan afgelezen worden dat 31 studenten een goede werking van het project ervaarden. Dit wil niet zeggen dat de goede werking persé tot stand komt dankzij Scrum, maar wel dat er een goede werking kan zijn wanneer Scrum gebruikt wordt. Er waren 4 studenten die een matige werking ervaarden en geen enkele respondent die een slechte werking ervaarde. De meest voorkomende opmerking over het voltooien van het project was dat de studenten hun idee van het eindproduct al vaak hadden moeten aanpassen waardoor ze terug een paar stappen achteruit stonden.

Vervolgens werd er gevraagd aan de 35 respondenten om aan te duiden van welke Scrum-elementen/handelingen ze gebruik maakten of zichtbaar waren gedurende het project. Dit was om te achterhalen of de studenten wel degelijk volgens Scrum werkten. Deze elementen/handelingen bestonden uit de elementen/handelingen die in Scrum voorkomen (deze verschillende elementen/handelingen zijn terug te vinden in bijlage 3). In figuur 14 zijn de aantallen per genummerd element terug te vinden. 29 van de 35 respondenten hadden minimaal 10 van de 14 elementen/handelingen aangeduid. Dit duidt erop dat ze wel degelijk volgens Scrum probeerden te werken maar dat er enkele elementen/handelingen zijn die door bepaalde groepen niet in acht werden genomen. De meest voorkomende elementen/handelingen die niet werden aangeduid waren de productbacklog (Nummer 6) en het werken met een tool (om zo burndown charts te kunnen maken; nummer 12) Ook de sprint retrospective ontbrak in 13 van de gevallen (nummer 9) en het bestaan van de groep uit gemotiveerde individuen werd ook in 12 van al de gevallen niet aangeduid. Verder was er in 10 gevallen geen sprake van een leidend persoon die ervoor zorgt dat alles goed verloopt (wat kan vergeleken worden met de scrumMaster). Er was ook in 10 gevallen geen sprake van een soort 'daily' Scrum meeting. Dit laat zien dat de Scrum-methode niet volledig werd toegepast, maar wel in grote mate. Ook dit is een risico wat voorkwam in de literatuurstudie. Wanneer de artefacten van de methode onvoldoende worden geïmplementeerd, kan dit leiden tot een minder goed beeld over de staat van het project op een gegeven moment. Dit kan dus leiden tot problemen die te laat worden gedetecteerd. Ook hier weer is het belangrijk dat de studenten alle elementen aangeleerd krijgen en deze ook werkelijk implementeren en deze zo vaak mogelijk updaten waardoor de staat van het project op elk gegeven moment bekeken kan worden.



Figuur 14: Aantal aangeduide elementen/handelingen van Scrum

Hierna is er ook gekeken naar de verbanden tussen de verschillende elementen/handelingen die voorkomen in de projectwerken van de studenten. De reden waarom dit gedaan werd, was om te weten te komen of het dezelfde studenten waren die bepaalde elementen minder hadden aangeduid. Op deze manier kon er bijvoorbeeld achterhaald worden of het dezelfde mensen waren die de product backlog en het gebruik van tool zoals Trello niet hadden aangeduid.

Aangezien de resultaten binair stonden weergegeven, zijn deze veranderd naar 'wel' en 'niet' (aangeduid). Op deze manier konden deze variabelen beschouwen worden als categorische variabelen en kon er gebruik gemaakt worden van de Chi-kwadraattoets om te bepalen of twee variabelen (elementen) afhankelijk of onafhankelijk van elkaar zijn. Bij de Chi-kwadraattoets zegt de nulhypothese dat twee variabelen onafhankelijk van elkaar zijn. Als de bij de chi-kwadraat horende p-waarde lager is dan 0.05, kunnen we de nulhypothese verwerpen en stellen dat de variabelen afhankelijk zijn van elkaar. Vervolgens hebben we de elementen die iedereen had aangeduid (element 2 en 13) uit de analyse gelaten aangezien het geen nut heeft om andere elementen met deze te vergelijken. In tabel 3 staan de verschillende p-waardes weergegeven.

	1	3	4	5	6	7	8	9	10	11	12	14
1	0	0.862	0.00474	0.521	0.241	0.00474	0.1087	0.1869	0.1601	0.8028	0.16	4E-05
3		0	0.00474	0.521	0.38	0.7155	0.1087	0.1869	0.1601	0.8028	0.464	0.8028
4			0	0.179	0.443	0.3647	0.3134	0.5718	0.4817	0.0775	0.678	0.0775
5				0	0.331	0.3134	0.3439	0.8249	0.2155	0.4897	0.26	0.357
6					0	0.759	0.5892	0.7619	0.1814	0.2072	0.537	0.8335
7						0	0.000779	0.0057	0.06833	0.0775	0.482	0.0775
8							0	0.00834	0.0426	0.49	0.26	0.4897
9								0	0.00034	0.6984	0.736	0.6984
10									0	0.6297	0.113	0.6297
11										0	0.293	0.7199
12											0	0.0438
14												0

Tabel 3: De berekende p-waardes van de chi-kwadraattoets

Wat opvalt is dat er tien p-waardes in het blauw zijn aangeduid. De bijhorende elementen zijn volgens deze berekeningen afhankelijke elementen. Het is alleen interessant om de verbanden tussen elementen te achterhalen die door de studenten minder zijn aangeduid. Dit omwille van het feit dat het logisch is dat diegene die veel zijn aangeduid door bijna allemaal dezelfde mensen zijn aangeduid. De elementen die dus interessant zijn om te vergelijken met elkaar zijn element 5, 6, 8, 9, 10 en 12. Als er naar tabel 3 gekeken wordt, zijn er drie koppels van elementen die in het groen gemarkeerd zijn (de andere koppels zijn in het rood gemarkeerd) die afhankelijk zijn van elkaar. Element 8 stelt de korte vergaderingen voor die gedaan moeten worden wanneer er samengekomen wordt en waarin besproken wordt wat er gedaan is, wat er moet gebeuren en wat er beter kan. Element 9 stelt de sprint retrospective voor en element 10 vraagt of de hele groep bestaat uit gemotiveerde individuen. Volgens de chi-kwadraattoets zijn het ongeveer dezelfde mensen die deze drie elementen hebben aangeduid. Dit betekent ook dat het ongeveer dezelfde mensen zijn die de elementen niet hebben aangeduid.

Wanneer minimaal 10 Scrum elementen/handelingen voorkwamen in een project van een respondent, werd gevraagd of ze goede ervaringen hadden met het toepassen van de Scrum-methode en of deze methode volgens hen voor betere productiviteit zorgt dan wanneer er gewerkt wordt met één finale deadline. In de veronderstelling dat enkele studenten de vraag foutief hebben gelezen, zijn er enkelen die minder dan 10 vakjes hadden ingevuld en alsnog deze vraag hebben beantwoord. 29 van de 31 respondenten hadden "ja" geantwoord op deze vraag en slechts 2 studenten hadden "nee" geantwoord. De opmerkingen die erbij werden vermeld waren zeer positief. Zo waren de volgende opmerkingen de meest voorkomende bij de studenten die "ja" hadden geantwoord:

- "Problemen worden vroegtijdig gevonden zodat er snelle aanpassingen kunnen gemaakt worden"
- "De tussentijdse deadlines zorgen ervoor dat de workload wordt verspreid"
- "Minder last van uitstelgedrag"

De twee studenten die "nee" antwoordden hebben de volgende opmerkingen erbij genoteerd:

- "Scrum is een goede methode voor software projecten. In deze context zorgt het alleen voor meer administratief werk."
- "Niet voldoende werk dat gebruikte methode een verschil maakt. Meer werk nodig om voordelen van scrum op te merken."

Hieruit blijkt dat enkele studenten vinden dat het werken volgens Scrum wanneer er niet voldoende werk is, zorgt voor onnodige werklast die geen meerwaarde biedt en er beter volgens een traditionele aanpak gewerkt kan worden.

In geval er minder dan 10 Scrum-elementen/handelingen voorkwamen, werd er gevraagd wat de reden was voor het ontbreken hiervan. Deze vraag werd door negen respondenten ingevuld en de meest opmerkelijke opmerkingen zijn de volgende:

- "Het formeel noteren van alles wat er moet gebeuren in de sprint is vaak niet nodig, aangezien het aantal taken per sprint beperkt is. Het bespreken van afgesloten sprints wordt

volgens mij niet gedaan omdat iedereen gewoon aan de volgende taken wilt beginnen en omdat alles grotendeels vlot verloopt.”

- “Over een burndown chart is nooit iets gezegd geweest. De artefacten die wij moeten afleveren zijn compleet anders iedere sprint. De voorgaande artefacten zien dus geen nieuwe iteratie meer.”
- “We zijn tot nu toe nog niet in een stage gekomen waar we de noodzaak en het nut van die backlog zagen.”
- “Dat is te strikt/te veel administratie. Wees duidelijk en kort en vertrouw teammates in wat ze doen.”
- “Gebrek aan kennis en motivatie bij medestudenten. De deliverables zijn vaak onduidelijk tijdens sprints. De agile manier van werken wordt omwille van heel verschillende opleidingen niet volledig gehanteerd.”
- “We gebruiken Trello met een to do/doing/done, maar niet als grafische voorstelling”
- “Ik denk dat iedere groep zijn eigen draai moet vinden. Op het begin kan je je vastklikken aan een principe, maar in de praktijk moet de groep zelf kijken welke zaken van een principe wel of niet voor hen werken.”
- “Niet iedereen kan gemotiveerd zijn voor alles. Trello wordt geforceerd gebruikt, in dit project eigenlijk niet nodig. Alles wordt door iedereen gecontroleerd.”
- “In principe zijn alle elementen in zekere mate toepasbaar op het project”

Ook hier weer wordt aangehaald dat er te weinig werk was om volledig volgens Scrum te werken. Wat opvalt is dat niet alle elementen van Scrum worden gebruikt omdat het onderwijsteam dit ofwel niet heeft aangehaald ofwel niet grondig heeft toegelicht waardoor de studenten niet volledig volgens Scrum ‘kunnen’ werken. Dit is zeker het geval voor de ‘product backlog’, waarin de features die gedaan moeten worden vermeld staan. De product backlog is een must bij het gebruik van Scrum. In de context van dit opleidingsonderdeel kan het wel zijn dat de studenten elke week of twee weken taken krijgen die ze moeten voltooien wat dan ook als een soort van product backlog kan beschouwd worden. Een tool zoals Trello wordt gebruikt als een hulpmiddel. Trello⁴ is een op web gebaseerde projectmanagement applicatie waarbij er visuele management borden kunnen worden gecreëerd. In het geval dit niet wordt gebruikt betekent dit echter niet dat de manier van werken niet als Scrum gezien kan worden. Verder is motivatie bij sommige studenten ook een probleem. Ongemotiveerde individuen zorgen ervoor dat het project geen goede vordering maakt omdat het werk enerzijds misschien niet goed wordt gedaan en anderzijds de werklast op de schouders van de andere teamleden komt te liggen.

Tot slot werd er gevraagd of deze manier van werken volgens hen toepasbaar is op alle types van projecten. 21 respondenten meenden van wel en 14 respondenten meenden van niet. Enkel van de 14 respondenten beweerden dat dit niet toepasbaar is op kleine individuele projecten of projecten met twee personen met vaste deliverables. Hierbij kan er volgens hen beter gebruik gemaakt worden van de waterval-methode, volgens de student. Een respondent vertelde dat Scrum moet gebruikt worden bij projecten waar heel duidelijke deelresultaten gedefinieerd kunnen worden. Verder zijn

⁴ <https://trello.com/>

projecten met een korte tijdsperiode volgens de respondenten niet ideaal om volgens Scrum te werken. Ook hier gaf een respondent de opmerking: "Projecten waarvan de effectieve werkuren veel hoger liggen, kunnen de overhead van 'full scrum' aan. Ons project heeft weinig gepresteerde uren." Dit duidt op het feit dat er te weinig werk was bij dit opleidingsonderdeel. Een persoon die "ja" antwoordde gaf als opmerking dat het toepasbaar is op alle types van projecten mits het aangepast wordt naar de noden en de setting van het project.

3.3 De principes van agile projectmanagement toegepast op het hoger onderwijs

Vervolgens zullen de twaalf principes van agile projectmanagement eerst geformuleerd worden zoals ze in het agile manifesto worden beschreven, gevolgd door een aanpassing, die gebaseerd is op de bevindingen in de literatuurstudie, naar het hoger onderwijs. Deze aanpassing wordt besproken aan de hand van de bekomen resultaten uit de enquêtes om hieruit af te leiden of het desbetreffende principe toepasbaar is in het hoger onderwijs.

Principe 1: "Onze hoogste prioriteit is het tevreden stellen van de klant door het vroegtijdig en voortdurend opleveren van waardevolle software."

Onderwijscontext: Er moet gewerkt worden met tussentijdse deadlines van afgewerkte deeltaken (features uit de product backlog) van een project, die gecontroleerd worden door het onderwijsteam. Op deze manier kan er vroegtijdig feedback gegeven worden waardoor het eindproduct waardevoller is dan wanneer er gewerkt wordt met één finale deadline.

De tussentijdse deadlines slaan op het feit dat er, zoals in de literatuurstudie werd vermeld, bij APM gewerkt wordt met kleine releases. Deze kleine releases zorgen ervoor dat problemen vroegtijdig gevonden worden en dit zo snel mogelijk kan aangepast worden zonder dat er veel stappen terug gegaan wordt. Ook zorgt dit voor een reducering in planningsfouten en kan er makkelijker ingespeeld worden op de veranderende behoeften van de klant (de docent in dit geval).

In de inleidende enquête werd gevraagd aan de respondenten of ze liever tussentijdse deadlines of één finale deadline hebben. Het opvallende is dat er 32 van de 43 studenten tussentijdse deadlines verkiezen. De redenen die hiervoor werden gegeven zijn de volgende:

- vermindert uitstelgedrag;
- tijdige feedback en referentiepunten voor progressie van het project;
- verspreidt de werkdruk;
- voorkomt het falen van een project, en;
- betere prestaties onder druk voor een deadline.

De 11 studenten die één finale deadline verkozen hadden als redenen:

- tussentijdse deadlines kunnen conflicten veroorzaken met andere deadlines;
- voorkeur voor zelf plannen, en;
- moeilijk combineerbaar met een druk leven buiten school.

Er kan geconstateerd worden dat een meerderheid voorkeur heeft voor tussentijdse deadlines. Dit zorgt voor druk waardoor ze tijdig aan het project werken en kunnen ze een project inleveren van betere kwaliteit doordat de regelmatige feedback van het onderwijsteam hen wijst op wat beter of

anders kan. De minderheid, die voorkeur heeft voor één finale deadline, bepaalt liever zelf wanneer ze aan het project werken met als reden dat ze andere deadlines hebben en een druk leven leiden waardoor ze volgens hen niet altijd tijd hebben om aan een project te werken. Wanneer een student studeert aan een universiteit of hogeschool is het zo dat er een bepaalde studiebelasting van de student per opleidingsonderdeel verwacht wordt waardoor een druk leven buiten school niet bepaald een goed argument is.

Om te achterhalen wat de ideale hoeveelheid werkdruk is voor een opleidingsonderdeel werd er in de tweede enquête gevraagd aan de studenten wat zij als ideale sprintlengte beschouwen. 20 studenten waren van mening dat twee weken ideaal is. Zo werd er aangehaald dat dit goed is voor teams van vijf personen en dat dit enerzijds niet te kort is maar anderzijds ook niet te lang. Verder waren er 13 respondenten die vonden dat twee tot drie weken ideaal is afhankelijk van het project en het werk dat er verricht moet worden. Ook werd er aangehaald dat de werkdruk bij sprints van één week te hoog kan zijn in combinatie met andere vakken. Dit kan gerelateerd worden aan de conflicten met andere deadlines wat aangehaald werd in de inleidende enquête. Vervolgens vertelde een student dat er bij sommige opdrachten, zoals het afnemen van interviews, meer tijd nodig is, maar dat de sprints korter mogen zijn wanneer het kleine deeltaken zijn. Deze persoon beweert dus dat de lengte van de sprints moet afhangen van het te verwezenlijken werk. Eén persoon was van mening dat de ideale sprintlengte tussen de twee en vier weken ligt met ook hier weer de opmerking dat het afhangt van het type project en het te verwezenlijken werk. Tot slot was er één respondent die vond dat drie à vier weken de ideale lengte van een sprint is, zonder verdere uitleg.

Uit de resultaten van deze twee analyses kan er geconstateerd worden dat werken met sprints van twee weken als ideaal wordt gezien, maar dat hiervan afgeweken kan worden wanneer de werkdruk hoger of lager ligt. De tussentijdse deadlines kunnen beschouwd worden als de deeltaken/deelproducten (features uit de product backlog) die voltooid moeten zijn na een sprint. Er kan dus geconcludeerd worden dat principe 1 toepasbaar is op het hoger onderwijs, aangezien de kwaliteit van het eindproduct hoog zal zijn, mits er rekening wordt gehouden met hetgeen hierboven werd geconstateerd. Verder zal de klant, namelijk het onderwijsteam, appreciatie hebben voor de studenten aangezien er goed werk geleverd wordt.

Principe 2: “Verwelkom veranderende behoeftes, zelfs laat in het ontwikkelingsproces. Agile processen benutten verandering tot concurrentievoordeel van de klant.”

Onderwijscontext: In het hoger onderwijs is het vaak zo dat een project een opdracht met een vaste einddeliverable is, die niet verandert gedurende het project. Om de studenten klaar te stomen voor een steeds veranderende omgeving in het bedrijfsleven, kunnen eventueel veranderende elementen in de opdracht verwerkt worden waardoor de studenten hierop moeten inspelen.

Ter onderbouwing van dit principe zijn er geen specifieke vragen gesteld in de enquêtes, hoewel er wel veranderende elementen voorkomen in het opleidingsonderdeel. Uit de literatuurstudie is gebleken dat de noden van klanten in een bedrijfscontext steeds kunnen veranderen. Dit omdat er een product ontwikkeld wordt voor de klant en dit aan de wensen van de klant moet voldoen.

Er kan een onderscheid gemaakt worden tussen twee types van projecten. Enerzijds, zoals hierboven reeds vermeld werd, is de einddeliverable van een project vaak iets wat op voorhand door het onderwijsteam gekend is. Anderzijds, zijn er projecten waarbij studenten iets moeten uitvinden of ontwikkelen dat bepaalde elementen (uit de les) moet bevatten, maar waarbij de einddeliverable uniek is. Voor het eerste type project kan het onderwijsteam zelf veranderende elementen toevoegen waarop de studenten moeten inspelen. Bij het tweede type project, zoals het opleidingsonderdeel waarbinnen deze enquête werd afgenomen, komen de veranderingen gedurende het project vanzelf aan het licht of zorgt de feedback van het onderwijsteam ervoor dat de studenten veranderingen moeten doorvoeren waardoor ze in een iteratief proces terecht komen. Het nadeel dat hieraan verbonden kan zijn, is dat er frustraties kunnen optreden bij de studenten omwille van het 'nutteloze' werk dat ze eerder gedaan hebben. Dit werd ook aangehaald bij de vraag over de werking van het project. Hier waren er studenten die hun idee deels hadden uitgewerkt maar wijzigingen, zoals het veranderen van hun idee, moesten doorvoeren waardoor ze een stap of enkele stappen achteruit gingen in het voltooien van het project.

Er kan vanuit gegaan worden dat principe 2 toepasbaar is in het hoger onderwijs aangezien veranderingen enerzijds automatisch gebeuren in een project waar volgens een agile methode gewerkt wordt en dit anderzijds door het onderwijsteam kan geïmplementeerd worden.

Principe 3: "Lever regelmatig werkende software op. Liefst iedere paar weken, hooguit iedere paar maanden."

Dit principe hangt nauw samen aan principe 1. Alleen zal de focus, in een niet software-gerelateerd opleidingsonderdeel, liggen op het regelmatig opleveren van een 'werkend' deelproduct dat als 'done' beschouwd kan worden. Een werkend deelproduct zou kunnen betekenen dat het aan bepaalde criteria moet voldoen. Zoals in de literatuurstudie werd aangehaald moet er gedefinieerd worden wanneer een feature als 'done' beschouwd kan worden. Dit betekent dat het dus moet voldoen aan de vooropgestelde criteria. De feedback van het onderwijsteam kan zich toespitsten op deze criteria, waardoor het team na de aanpassing het deelproduct als 'done' kan beschouwen.

Principe 4: "Mensen uit de business en ontwikkelaars moeten dagelijks samenwerken gedurende het gehele project."

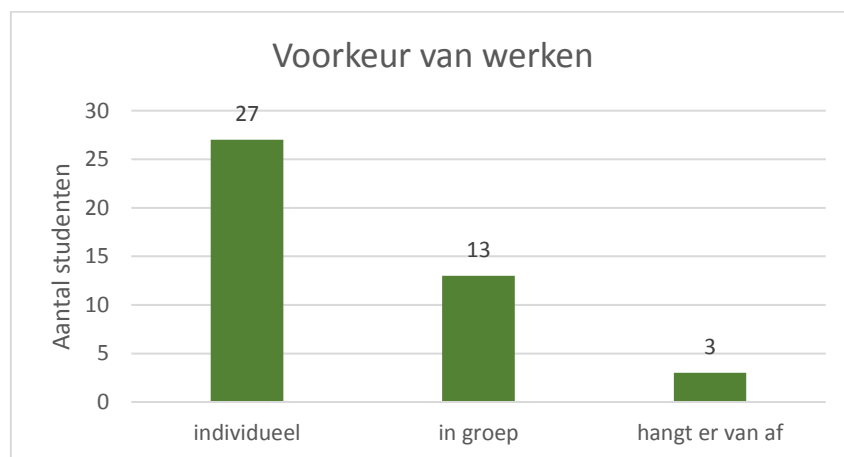
Onderwijscontext: Studenten moeten op regelmatige basis met elkaar samenwerken gedurende het gehele project en moeten regelmatig feedback krijgen van het onderwijsteam om aan de noden van het onderwijsteam te voldoen.

Het probleem dat zich hier voordoet en ook werd aangehaald in de literatuurstudie is dat het onmogelijk is om studenten dagelijks samen te laten komen en dat het onmogelijk is om van studenten te verwachten dat ze iedere dag aan een project werken. Bij principe 1 werd er al aangehaald dat de ideale lengte van een sprint min of meer twee weken is. Dit betekent dat een bepaald feature of features in die tijd klaar moeten geraken. De vraag die hierbij gesteld kan worden is: 'Hoe vaak moeten de studenten samenkomen om aan het project te werken?'. Aangezien bij agile projectmanagement collaboratie tussen de teamleden en de klant centraal staat, wordt in de onderwijscontext aangeraden dat studenten minimaal twee keer per week samenkomen om aan het project te werken. Bij het opleidingsonderdeel waar de enquête is afgenomen zijn er steeds twee

momenten in de week waarop de studenten onder toezicht van het onderwijsteam aan het project werkten. Dit betekent dat de studenten per sprint vier momenten van telkens twee uur hadden om gezamenlijk aan het project te werken, en bij deze momenten ook feedback van het onderwijsteam konden vergaren.

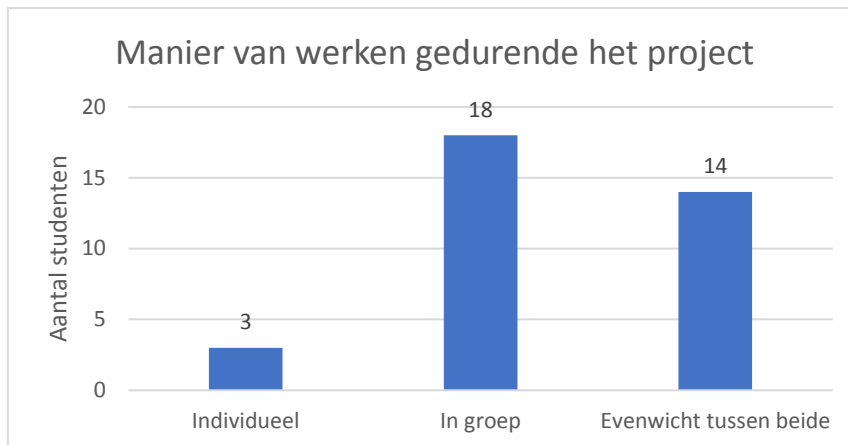
In de inleidende enquête werd gevraagd aan de studenten of ze het liefst individueel of in groep aan een project werken. De resultaten worden in figuur 15 weergegeven. 27 van de respondenten vertelden dat ze het liefst individueel aan een project werken omdat ze van mening zijn dat wanneer ze zelfstandig werken, ze sneller en effectiever werken. Verder kunnen ze op deze manier beter plannen omdat ze met niemand rekening hoeven te houden. Ook gaven ze als opmerking dat er meer afleiding is wanneer er in groep gewerkt wordt en dat er vaak veel tijd genomen wordt bij beslissingen omtrent iets kleins, waardoor het proces in het algemeen vertraagd wordt. Dit kan te wijten zijn aan het feit dat de opdracht niet uitdagend genoeg is of dat de motivatie ontbreekt bij de studenten. De studenten beweren dat een project opsplitsen in rollen en deeltaken die verdeeld kunnen worden de meest effectieve manier van werken is. Op deze manier kunnen ze samenkomen om alles te bespreken en de delen samenvoegen wanneer elk teamlid zijn of haar deel klaar heeft.

De 13 respondenten die verkieszen om in groep te werken beweerden dat er in groep tot meer inzichten gekomen kan worden dan wanneer er alleen wordt gewerkt en dat de ervaring van de groep gebruikt kan worden om bepaalde problemen aan te pakken. Bij het werken in een groep is er volgens enkelen sprake van "peer pressure" waardoor er goed aan het project wordt doorgewerkt, er bij problemen altijd directe communicatie mogelijk is en er constante controle is op de teamleden. Verder vinden sommige respondenten het interessanter om in groep te werken wanneer er een goede werksfeer heerst. Tot slot zijn er 3 studenten die van mening zijn dat het afhangt van het type project, de projectgrootte en de moeilijkheidsgraad.



Figuur 15: De voorkeur voor de manier van werken aan een project

Vervolgens werd er in de tweede enquête gevraagd op welke manier ze werkelijk werkten aan het project. Zoals te zien is in figuur 16, werkten 18 studenten alleen maar in groep, slechts 3 studenten individueel en 14 studenten die een mix van beide hanteerden. De respondenten die een evenwicht tussen beide gebruikten, vertelden dat ze in groep alles bespreken, de groepstaken samen maken, de taken dan verdelen, hier thuis individueel aan werken en deze achteraf terug in groep bespreken en verbeteren.

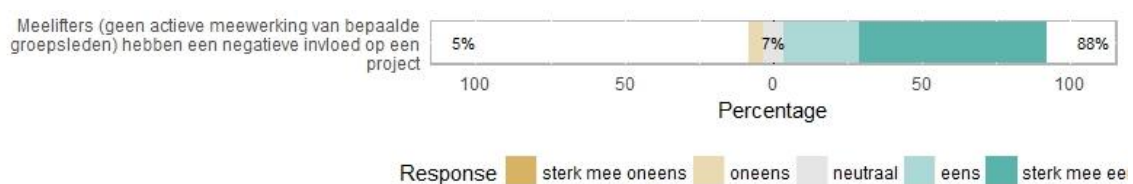


Figuur 16: De manier van werken gedurende het project

Diegene die 'in groep' hadden aangeduid, hadden in de meeste gevallen ongeveer dezelfde redenering als de voorgaande. Zo waren er enkelen die vertelden: "Alles wordt in groep gedaan maar als er kleine deeltjes afgemaakt moeten worden, worden deze thuis individueel gedaan." Ook hier kwam de opmerking weer dat er te weinig werk was om de taken te verdelen om er thuis individueel aan te werken wat slaat op een te lage werklast per sprint. Van de drie personen die voor 'individueel' hadden gekozen had een student als reden gegeven dat het moeilijk is om samen te komen door verschillen in agenda's en uurroosters tussen de verschillende teamleden. De twee anderen vertelden dat tijdens de les besproken wordt wie wat doet en dit vervolgens individueel thuis wordt gedaan.

Deze bevindingen tonen aan dat de momenten dat de studenten samen kunnen werken, ervoor zorgen dat deze optimaal gebruikt worden om de taken te voltooien. Aangezien Scrum opteert voor zoveel mogelijk face-to-face communicatie, is het inplannen van vaste contactmomenten zeker een pluspunt voor het project. Het neemt niet weg dat studenten liever individueel werken, maar deze vaste contactmomenten verplichten de studenten om face-to-face samen te werken waardoor hun voorkeur in de toekomst kan wijzigen naar het 'in groep' werken omdat ze de voordelen ervan kunnen inzien. Hieruit kan dus worden afgeleid dat wanneer er volgens de principes van Scrum gewerkt wordt, dit het werken in groep zou kunnen bevorderen.

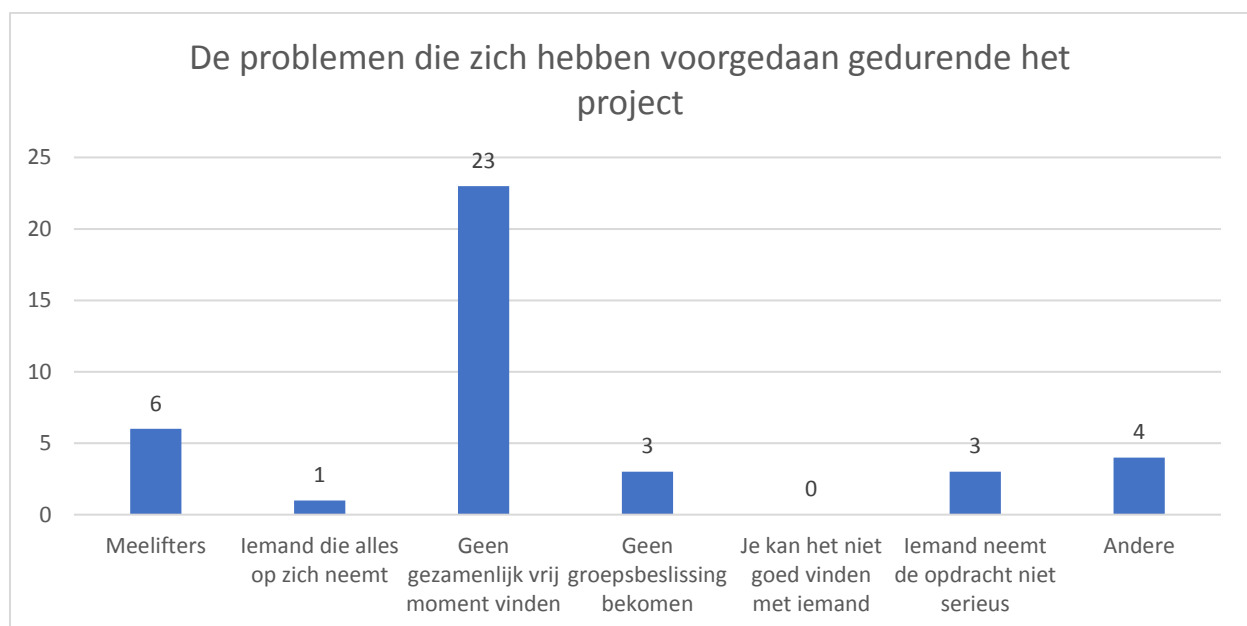
Voordat er werd gevraagd aan de studenten welke problemen er zich al hadden voorgedaan tijdens het project, werd er in de inleidende enquête gevraagd of studenten denken dat meelifers een negatieve invloed hebben op het maken en voltooien van een project. Dit werd bevestigd aangezien 88% van alle studenten het er mee eens of sterk mee eens was. Dit resultaat is af te lezen in figuur 17. De grootte van de kleur in de balk toont het percentage aan bij één van de bijhorende antwoorden.



Figuur 17: De mate waarin meelifers een negatieve invloed hebben op een project

Vervolgens werd er aan de studenten in de tweede enquête gevraagd welke problemen er zich al hadden voorgedaan gedurende het project. De resultaten staan voorgesteld in figuur 18. Hieruit is duidelijk dat, aangezien het opleidingsonderdeel wordt gevolgd door studenten uit verschillende opleidingen met verschillende uurroosters, het vinden van een gezamenlijk vrij moment voor problemen zorgde. Door de verschillende lessenroosters is het vaak moeilijk voor twee verschillende opleidingen om een gezamenlijk vrij moment te vinden. Dit toont aan dat cross-functionele teams vaak een moeilijk iets is om te verwezenlijken in een hoger onderwijscontext. Als er verschillende opleidingen een opleidingsonderdeel samen volgen en er een project bij komt kijken, is het aangeraden dat de agenda's van deze twee opleidingen op elkaar worden afgestemd. Natuurlijk is het wel zo dat binnen een opleiding ook verschillende kwaliteiten zitten waardoor cross-functionele teams ook mogelijk zijn. Verder zijn er enkele meelifers die profiteren van het harde werk van anderen en zijn er sommigen die de opdracht niet serieus nemen. Dit kan invloed hebben op de motivatie van de harde werkers aangezien zij meer werk hebben dan wanneer het werk op een eerlijke manier verdeeld wordt. Enkele studenten kampten ook met het feit dat ze niet altijd komen tot het nemen van groepsbeslissingen. Dit laatste kan ook invloed hebben op de motivatie aangezien het kan zijn dat wanneer er iets beslist wordt en één van de studenten hier niet achter staat, deze minder gemotiveerd zal zijn om zich volledig in te zetten. Motivatie zal verder worden besproken in principe 5. De 4 studenten die 'andere' hadden aangeduid, hadden als opmerking dat er vaak teamleden te laat komen of niet komen opdagen. Ook dit kan ervoor zorgen dat de motivatie binnen een team verslechtert.

Als aanvulling hierop werd er gevraagd op welke manier de studenten met deze problemen zijn omgegaan. Voor het vinden van een gezamenlijk vrij moment werd geen concrete oplossing aangereikt. Wel werd aangehaald dat in dringende gevallen de beschikbare groepsleden samenkwamen om te werken aan het project, zonder dat ieder teamlid aanwezig kon zijn. In andere gevallen werden teamleden aangesproken op hun gedrag.



Figuur 18: De problemen die zich tijdens het project hebben voorgedaan

Uit al het voorgaande kan er geconstateerd worden dat elke dag samenwerken een onmogelijk iets is voor studenten in de huidige organisatievorm van hoger onderwijs en dat dit voor een project van een opleidingsonderdeel ook niet nodig is aangezien de studiebelasting hierop is afgestemd. Wanneer er volgens een agile methode gewerkt wordt, is het wel aangeraden dat er regelmatig fysiek wordt samengekomen. Dit kan op basis van vaste momenten die in het lessenrooster door het onderwijsteam worden ingepland, maar ook door het op elkaar afstemmen van de agenda's van studenten. Als er dringend een feature moet worden afgewerkt, maar er kan geen vrij moment gevonden worden, dan moeten diegenen die wel beschikbaar zijn samenkomen en kan er achteraf via een communicatietool alles gedeeld, bekeken en besproken worden. Er kan dus geconcludeerd worden dat principe 4 in zeker mate toepasbaar is wanneer het woord 'dagelijks' naar 'regelmatig' gewijzigd wordt.

Principe 5: "Bouw projecten rond gemotiveerde individuen. Geef hen de omgeving en ondersteuning die ze nodig hebben en vertrouw erop dat ze de klus klaren."

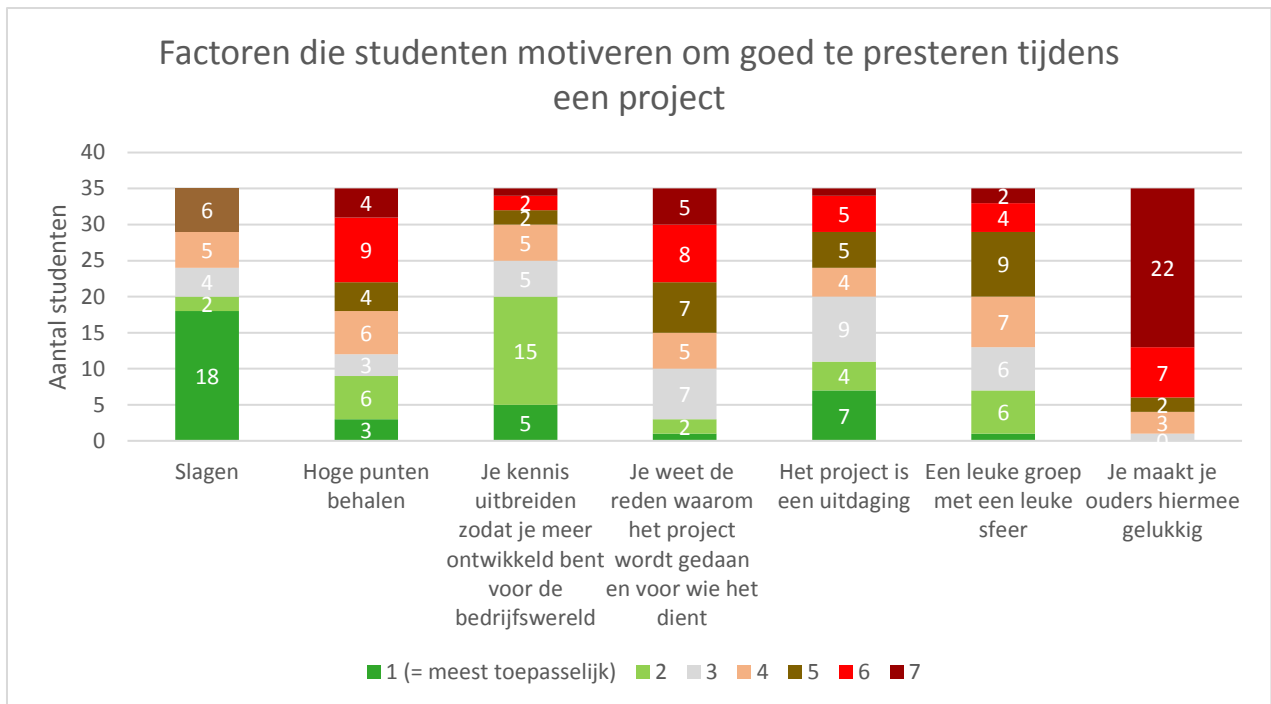
Onderwijscontext: Wanneer er in een opleidingsonderdeel een project gedaan wordt, moeten alle studenten van dit opleidingsonderdeel dit project maken. Echter bestaat er de kans dat niet iedereen even gemotiveerd is om zich volledig in te zetten voor het project. Er moet voor worden gezorgd dat de studenten op een bepaalde manier wel gemotiveerd zijn/worden. Dit kan bijvoorbeeld, zoals aangehaald in de literatuurstudie, bereikt worden door de studenten van een ondersteunende omgeving te voorzien, maar ook door strategieën toe te passen die waarde en verwachtingen bij studenten doen stijgen.

Om te achterhalen of de studenten extrinsiek gemotiveerd (instrumentele waarde) zijn om goed te presteren tijdens een project, werd er aan hen in de tweede enquête gevraagd de volgende stelling aan de hand van een vijfpuntige likert-schaal te beoordelen: "Een student is gemotiveerd om goed te presteren omdat hij/zij een goede score wil behalen". Zoals te zien is in figuur 19, is 69% van de studenten het er mee eens of sterk mee eens dat een goede score een motivator is om goed te presteren. 20% is hier neutraal over en 11% is het er niet mee eens of sterk mee oneens. Dit toont aan dat het merendeel van de studenten extrinsiek gemotiveerd is om goed te presteren.



Figuur 19: De mate waarin een goede score behalen de student motiveert

Om hier verder op in te gaan werd er in de tweede enquête gevraagd aan de studenten een ranking te maken van de factoren die hen motiveren om goed te presteren tijdens een project. Zo kon er worden nagegaan of de studenten ook intrinsieke waardes en/of eindwaardes hebben. De resultaten van deze ranking worden voorgesteld in figuur 20.

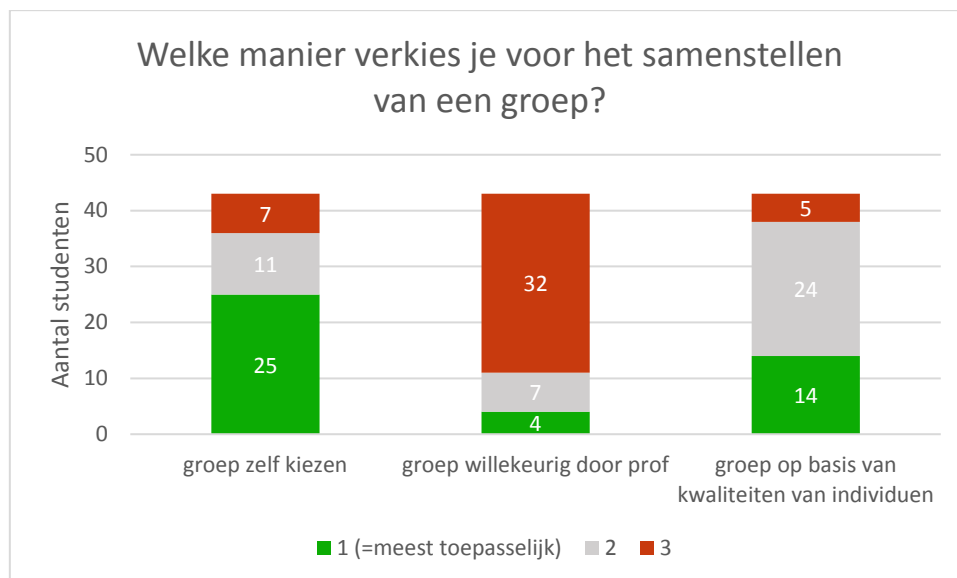


Figuur 20: Een ranking van de factoren die studenten motiveren goed te presteren tijdens een project

Hieruit is af te leiden dat slagen (instrumentele waarde) de grootste motivator is gevolgd door het uitbreiden van kennis (intrinsieke waarde) en een uitdagend project (eindwaarde aangezien een uitdagend project ook leidt tot voldoening wanneer het project afgewerkt is). Zoals aangehaald in de literatuurstudie, is het vaak zo dat de bronnen van verschillende waardes in combinatie voorkomen en elkaar aanvullen. Hoge punten behalen komt hier op de vierde plaats. Dit laat zien dat studenten slagen belangrijker vinden dan het behalen van een hoge score. De frequenties van de overblijvende mogelijke motivatoren zijn ongeveer evenredig verdeeld met uitzondering van het gelukkig maken van ouders. Dit is in 22 van de gevallen als laatste gekozen en in zeven gevallen als voorlaatste, maar nooit als eerste of tweede. Uit deze ranking kunnen we ook afleiden dat er op vlak van doelen enkele studenten het sociale doel hebben aangezien sommigen een leuke groep met een leuke sfeer hoog waarderen. Uit de bovenstaande resultaten kunnen we concluderen dat de elementen die gelinkt zijn aan bepaalde waarde, door de studenten hoog gewaardeerd worden waardoor ze hoogstwaarschijnlijk hun gedrag hier ook op zullen afstemmen. Dit wil zeggen dat ze genoeg waardes hebben om gemotiveerd te zijn. Zoals in de literatuurstudie werd aangehaald moeten deze 'positieve' waardes in combinatie met werkzaamheidsverwachtingen (het eigen kunnen van een student) en een ondersteunde omgeving voorkomen om pas echt van motivatie te kunnen spreken. Zoals in 2.8.5 werd aangehaald zijn er enkele manieren aangehaald die waarde en verwachtingen kunnen creëren. Met een ondersteunende omgeving wordt bedoeld dat de docent bevoorbeeld benaderbaar moet zijn, of dat de medestudenten willen helpen waar nodig. Verder kan er met een ondersteunende omgeving ook bedoeld worden dat er een levensechte business omgeving gecreeërd wordt, zoals het toewijzen van een lokaal aan een team en dit lokaal 'ombouwen' tot een vergaderruimte.

Vervolgens zijn er nog elementen die invloed kunnen hebben op motivatie, zoals de samenstelling van het team. Ten eerste werd er in de inleidende enquête gevraagd welke manier de studenten verkiezen voor het samenstellen van de teams. Vervolgens werden in dit opleidingsonderdeel de

groepen samengesteld aan de hand van een vragenlijst die was opgesteld door het onderwijsteam en die de studenten moesten invullen. Deze vragenlijst was bedoeld om de kwaliteiten van de individuen te bepalen en om op deze manier cross-functionele teams samen te stellen. In de tweede enquête werd er gevraagd of ze deze methode voor het samenstellen van groepen beter vinden dan het zelf kiezen van de teamleden.



Figuur 21: Een ranking van de methodes voor het samenstellen van groepen

In figuur 21 zijn de resultaten die in de inleidende enquête werden verzameld weergegeven. Het is duidelijk dat studenten verkieszen om de groep zelf samen te stellen. Op de tweede plaats komt het samenstellen van een groep op basis van kwaliteiten van individuen. Het willekeurig samenstellen van de groep door het onderwijsteam wordt duidelijk niet geliefd.

Bij de vraag of de studenten deze methode (samenstelling volgens een vragenlijst) beter vinden dan het zelf samenstellen van een groep, was het opmerkelijk dat 22 van de 35 respondenten "ja" antwoordden. Verder worden enkele opmerkingen van deze studenten voorgesteld:

- "Ik denk het wel, iedereen heeft andere kwaliteiten. Dus het is gemakkelijker om deze uit te spelen naargelang de situatie"
- "Zo leer je andere mensen kennen en omgaan met verschillende manieren van werken en denken. In het werkveld worden de teams ook samengesteld op basis van kwaliteiten."
- "Echter heb ik niet het gevoel dat onze groep optimaal samengesteld is."
- "Zo krijg je echt teams die elkaar aanvullen en werk je niet altijd met dezelfde mensen."
- "Minder 'vriendjespolitiek', meer professionele samenwerking, op elkaars zwaktes aanvullen."

Verder waren er 9 respondenten die vonden dat teams samenstellen op basis van kwaliteiten van individuen geen goede manier is. De meest opvallende commentaren zijn de volgende:

- "Mensen vullen bewust vragen fout in. Een goed team bestaat uit 1 bepaalde compositie en deze is niet altijd voor elke groep te vormen binnen een klas, daar een over-of onderaanbod van een bepaald type mensen."

- "Zelf samenstellen laat toe om te werken met mensen die je goed kent (sterktes/zwaktes)"
- "In dit geval is de samenstelling van de groep goed uitgedraaid, maar doorgaans kies ik liever zelf mijn teamgenoten."
- "Met zelf samenstellen ken je de groepsleden al, dus kan je vlotter starten."
- "Ik ken de kwaliteiten van de andere leden niet en kan hier dus niet op inspelen. Vandaar dat ik voor schoolprojecten liever werk met mensen waarvan ik weet wat ik aan hen heb."

Het is duidelijk dat een groot deel van de studenten deze manier een goede manier vinden en open staan om met mensen die ze niet kennen samen te werken. Verder hebben diegene die "nee" hebben geantwoord goede argumenten aangehaald. Het is inderdaad zo dat wanneer er van de groepsleden geweten is wat de kwaliteiten zijn, er direct aan de slag gegaan kan worden en de taken goed verdeeld kunnen worden. Een overaanbod van een bepaalde kwaliteit kan zeker ook gebeuren, waardoor sommige groepen van bepaalde kwaliteiten te weinig hebben. Op deze twee problemen kan er door het onderwijsteam ingespeeld worden. Enkele voorstellen die zouden kunnen worden doorgevoerd:

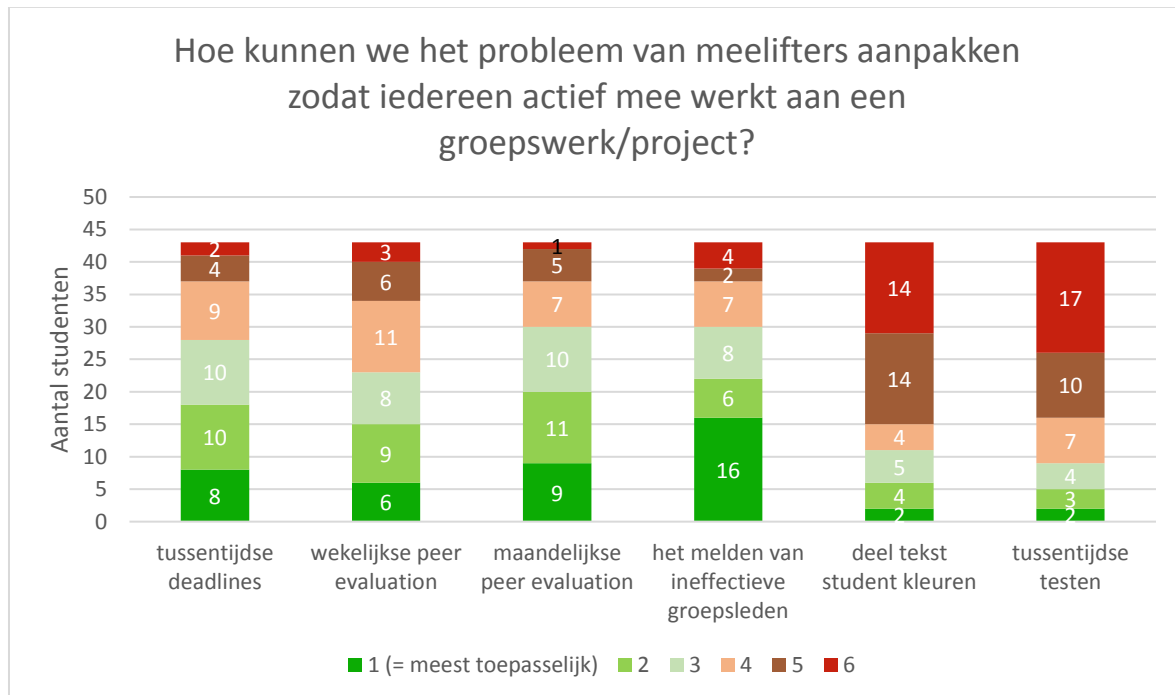
- Na het samenstellen van de teams op basis van kwaliteiten een profiel maken waarin de sterktes en zwaktes beschreven staan en deze afdrucken en uitdelen aan de groepsleden zodat ze het nieuwe team beter kunnen 'leren kennen'.
- Er proberen voor te zorgen dat wanneer er een mix van studenten van verschillende opleidingen éénzelfde opleidingsonderdeel volgen, de groepen minstens 1 of meer studenten van elke opleiding bevat (wat bij dit opleidingsonderdeel het geval al is).

Tot slot zijn er 4 studenten die voor 'neutraal' hadden gekozen. Een van deze respondenten antwoordde hierbij dat het interessant is om met deze mensen samen te werken, maar dat deze persoon nog niet overtuigd was dat deze manier de beste was. De andere respondenten hadden hier geen mening over.

Verder kan de grootte van een groep ook een rol spelen bij de motivatie van studenten. Aangezien er in de literatuurstudie werd aangehaald dat de ideale groeps grootte van een agile team tussen vier en negen personen ligt, leek het interessant om te achterhalen wat de ideale groeps grootte is volgens de studenten bij een project in het hoger onderwijs. Deze vraag werd in de tweede enquête gesteld en hier werden verschillende antwoorden op gegeven. Het meest voorkomende antwoord was vier groepsleden, maar volgens veel respondenten hangt de ideale groeps grootte af van het type project. Zo zijn er opdrachten die met twee personen kunnen voltooid worden. Bij grote projecten zijn meer mensen nodig om de werklust te verdelen. Hierdoor is er meer groepsdruk om de minder actieve teamleden ook te betrekken en is de communicatie moeilijker dan bij kleinere groepen. Er zijn ook enkele studenten die opteren voor een ongelijk aantal zoals vijf, zodat er bij een beslissing altijd een meerderheid is.

Een ander probleem, dat ook al aangehaald werd bij principe 4, zijn meelifters, ook wel bekend als free-riders. Dit zijn individuen die op de achtergrond blijven, zo weinig mogelijk werk verrichten en van het harde werk van andere teamleden profiteren. Dit is een veel voorkomend probleem bij studenten. Om te weten te komen wat de studenten als beste oplossingen zien voor het aanpakken van het free-rider probleem, werd er in de inleidende enquête aan hen gevraagd om een ranking te

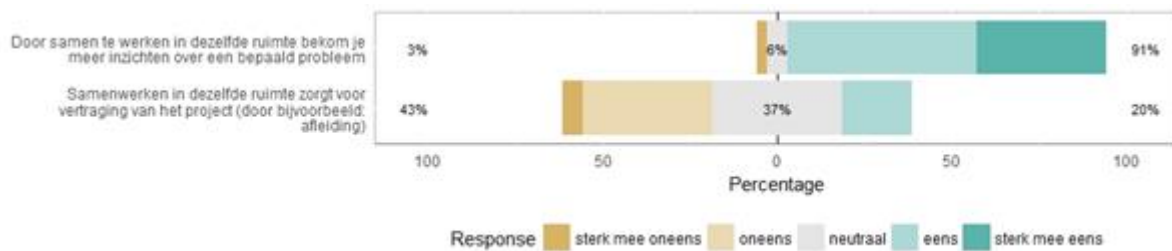
maken van de mogelijke oplossingen voor het aanpakken van meelifters. De resultaten zijn terug te vinden in figuur 22.



Figuur 22: Een ranking van de methodes voor het aanpakken van meelifters

Zoals af te lezen is in figuur 22, is het melden van ineffektieve groepsleden aan het onderwijsteam de grote winnaar, op vlak van de keuze voor het meest toepasselijke. Op deze manier kan het onderwijsteam acties ondernemen zodat deze persoon meer inzet toont. Dit wordt gevolgd door de maandelijkse peerevaluatie waar de ineffektieve groepsleden ook gemeld worden aan het onderwijsteam (zolang deze eerlijk worden ingevuld). Er is ook een deel van de respondenten die tussentijdse deadlines hoog in de rang plaatst. Dit zorgt ervoor dat er druk staat op de studenten waardoor ze allemaal verplicht worden om aan het project te werken. Een wekelijkse peerevaluatie volgt hierop, maar dit lijkt in de praktijk moeilijk te hanteren in het hoger onderwijs, aangezien er niet elke week even hard wordt gewerkt aan een project en de bekomen resultaten hierdoor verwaarloosbaar zouden zijn. Tot slot zijn het in kleur aanduiden van delen van een taak die een student heeft geschreven of gemaakt en tussentijdse testen niet geliefd bij de studenten.

In dit principe wordt aangehaald dat het team de nodige omgeving en ondersteuning moet krijgen om agile te kunnen werken. Dit kan, zoals eerder vermeld, bijvoorbeeld gedaan worden door een ruimte te voorzien waar de studenten samen kunnen werken. In de literatuurstudie werd er aangehaald dat het team samen met de klant zoveel mogelijk 'co-located' moet zijn. Tijdens de vaste contacturen is dit het geval aangezien het onderwijsteam zich in dezelfde ruimte bevindt als de teams. In de tweede enquête werden twee stellingen gegeven omtrent het werken in dezelfde ruimte. De resultaten van deze vijfpuntige likert-schalen worden voorgesteld in figuur 23.



Figuur 23: Stellingen over het samenwerken in dezelfde ruimte

De eerste stelling betreft het bekomen van meer inzichten bij het samenwerken in dezelfde ruimte. 91% van de respondenten is het eens of sterk eens dat dit voor meer inzichten zorgt over bepaalde problemen. Dit laat nogmaals zien dat samenwerken in groep bepaalde voordelen biedt.

De tweede stelling gaat over vertraging van het project door middel van in dezelfde ruimte te werken. Hier zijn de resultaten verdeeld maar met een meerderheid die het er oneens of sterk oneens mee is (43%). Ook valt hierop dat een groot deel (37%) een neutrale mening heeft. Deze twee percentages laten zien dat er studenten zijn die van mening zijn dat in groep werken iets positief is (wat ook aansluit bij de vraag over de werking van de groep). Tot slot zijn er 20% van de respondenten die het er mee eens zijn. Dit wijst waarschijnlijk op het feit dat er soms wel sprake is van afleiding of andere elementen die het project vertragen wanneer er in groep gewerkt wordt. Dit hangt ook samen met het probleem van meelifters die waarschijnlijk de grootste oorzaak zijn van afleiding. Als de oorzaak van de afleiding ligt bij bepaalde studenten die over andere zaken dan het project praten, is het aangeraden hen hierop te wijzen. Een andere oplossing is om regelmatig korte pauzes in te lassen zodat er over andere zaken gesproken kan worden. Tijdens de werkmomenten moet heel het team aandachtig zijn en met de zaak bezig zijn, zodat de beste prestatie wordt geleverd.

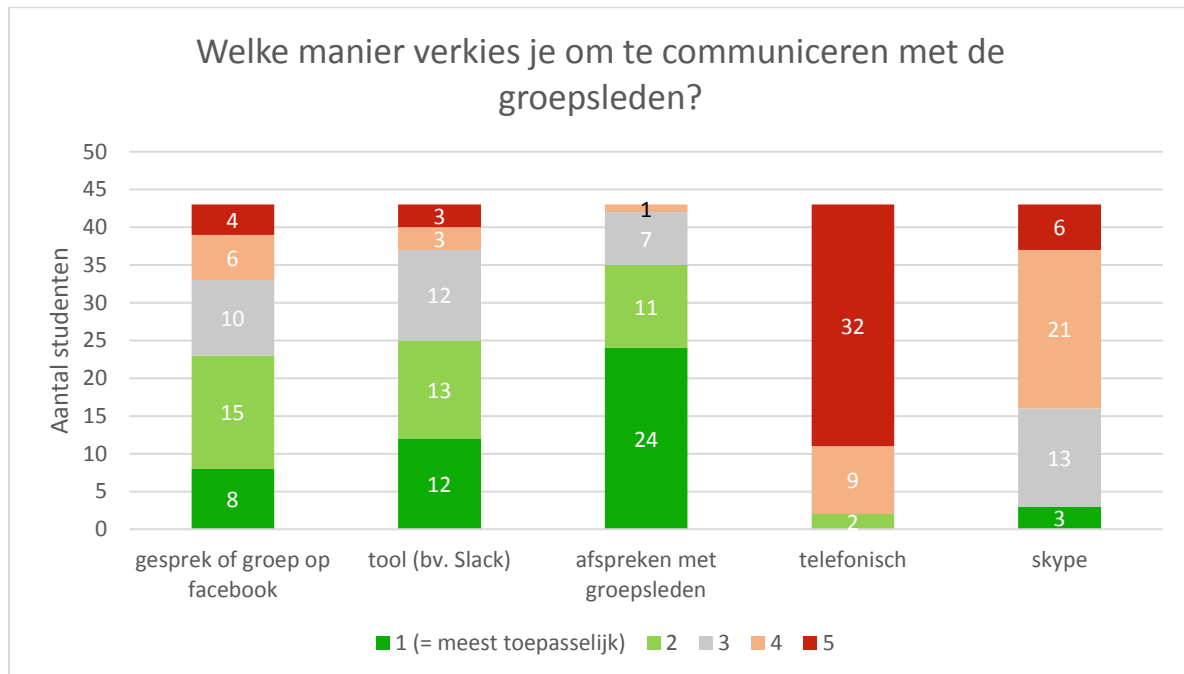
Uit al het voorgaande kan er geconcludeerd worden dat principe 5 toepasbaar is in het hoger onderwijs, mits de studenten allemaal gemotiveerd zijn of gemotiveerd worden door bepaalde maatregelen die er door het onderwijsteam worden getroffen en de nodige omgeving (een projectruimte) en benodigdheden (voldoende uitleg en feedback) door het onderwijsteam worden voorzien.

Principe 6: “De meest efficiënte en effectieve manier om informatie te delen in en met een ontwikkelteam is door met elkaar te praten.”

Onderwijscontext: Dit principe kan volledig overgenomen worden aangezien het communicatieaspect hier wordt aangehaald. Communicatie bij een project is zeer belangrijk en volgens de agile methodologie is face-to-face communicatie de beste manier van communiceren.

In de inleidende enquête werd er gevraagd aan de studenten een ranking te maken van de verschillende manieren van communicatie die ze verkiezen. Zo is in figuur 24 duidelijk dat een grote meerderheid verkiest om af te spreken met de groepsleden om face-to-face met elkaar te kunnen communiceren. Dit wordt gevolgd door het communiceren via een tool zoals Slack en het

communiceren via een gesprek of groep op Facebook. Slack⁵ is een op cloud gebaseerde verzameling van samenwerkingstools en -diensten die de communicatie tussen individuen of teams vergemakkelijkt. Skype en telefonische communicatie wordt door praktisch niemand erg gewaardeerd. Skype⁶ is een VoIP-programma van Microsoft, waarmee er over het internet getelefoneert kan worden.



Figuur 24: Een ranking over het verkieszen van manieren van communicatie

In de tweede enquête werd er gevraagd welke manier van communiceren ze werkelijk gebruikten gedurende het project. De resultaten worden voorgesteld in figuur 25 en komen grotendeels overeen met de voorkeur die de respondenten hadden bij de inleidende enquête. Dit bevestigt ook weer dat er voorkeur is om in groep te werken, maar dat er ook individueel gewerkt wordt en de deeltaken gedeeld worden in een tool zoals Slack of Facebook. Hier kunnen de studenten de deeltaken nakijken en aanpassen waar nodig. Sommige respondenten verkieszen een Facebook-groep/gesprek, maar dit hangt af van de voorkeur en eventuele gewoontes. Enkele studenten haalden aan dat ze Hangouts van Google gebruikten om met elkaar te communiceren en twee studenten die via Trello communiceerden. Google Hangouts⁷ is een platform voor instant messaging, IP-telefonie en videoconferentie ontwikkeld door Google. Dit laat zien dat er ook andere manieren van communiceren zijn waarvan het kan zijn dat sommige studenten geen weet hebben.

Vervolgens werd er gevraagd of dit volgens hen de meest efficiënte manier van werken is. Het meest voorkomende antwoord was dat ze de combinatie van werken met een tool en fysiek afspreken, de meest efficiënte manier van communiceren vinden. Iemand vertelde dat hij/zij liever met Slack werkt, maar dat de groepsleden hierop tegen waren. Een andere student vertelde dat hij/zij een voorkeur had om een Facebook-groep te gebruiken omdat dit een gewoonte is en hij/zij hierop veel

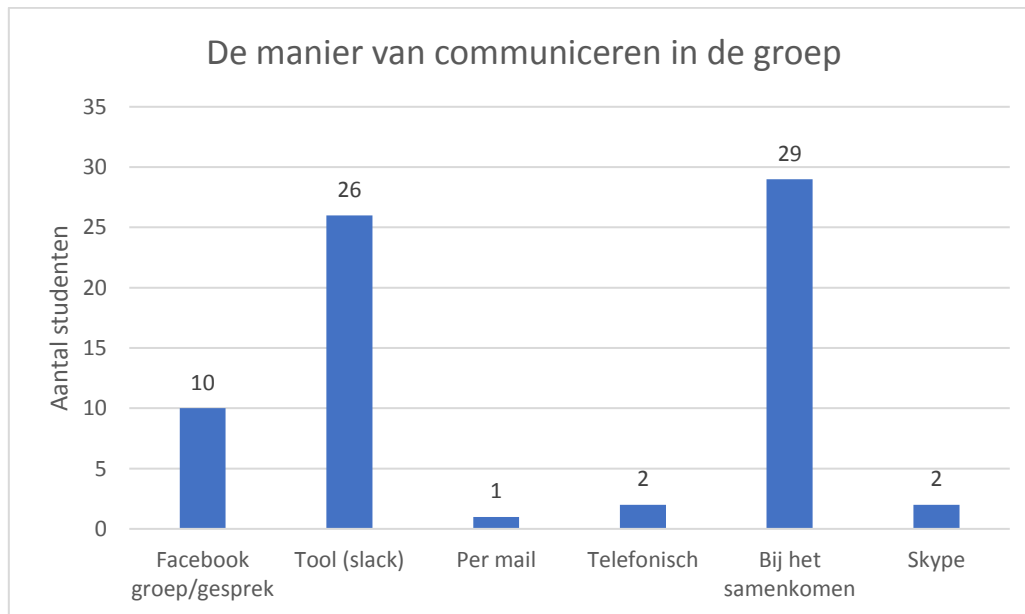
⁵ <https://slack.com/>

⁶ <https://www.skype.com/>

⁷ <https://hangouts.google.com>

actiever is. Dit toont ook aan dat er een goede overeenkomst moet zijn tussen de teamleden over de kanalen die ze gebruiken om te communiceren. De teamleden moeten zich aanpassen naar de voorkeuren van de meerderheid van de groep en wat het beste werkt voor de groep. Er kan in een team geëxperimenteerd worden aan het begin van een project om te kijken wat de meest effectieve en efficiënte manier is van communiceren.

Uit het voorgaande kan er geconcludeerd worden dat principe 6 toepasbaar is op het hoger onderwijs, mits er duidelijke afspraken gemaakt worden omtrent de manier van communiceren aan het begin van het project.



Figuur 25: De werkelijke manier van communiceren tijdens het project

Principe 7: “Werkende software is de belangrijkste maat voor voortgang.”

Onderwijscontext: Dit principe aangepast naar een project in het hoger onderwijs: “Een goede oplossing of benadering van de deeltaak van het project is de belangrijkste maat voor voortgang”

Dit hangt nauw samen met principe 1 waarbij er werd verteld dat de beste manier is om te werken met tussentijdse deadlines. Dit principe hangt ook samen met principe 3 aangezien het werkende software betreft. Door tussentijdse deadlines is de student verplicht om een feature te voltooien. Als de voltooiing van deze feature foutloos is, kan dit als ‘done’ beschouwd worden (zoals bij principe 3) en kan hier verder op gebouwd worden waardoor vooruitgang geboekt kan worden. Als het deelproduct niet foutloos is of niet aan de gewenste vooropgestelde criteria voldoet, kan het onderwijsteam hier constructieve feedback op voorzien zodat het deelproduct geoptimaliseerd kan worden. Als dit principe losgekoppeld wordt van het softwareontwikkelingsaspect is dit zeker een toepasbaar principe.

Zoals in de literatuurstudie aangehaald, moet de klant op tijd feedback geven en op tijd benaderbaar zijn. Dit wil dus zeggen dat de docent of het onderwijsteam ook steeds op tijd beschikbaar moet zijn om de studenten van feedback te voorzien, zodat de studenten op tijd de aanpassingen kunnen

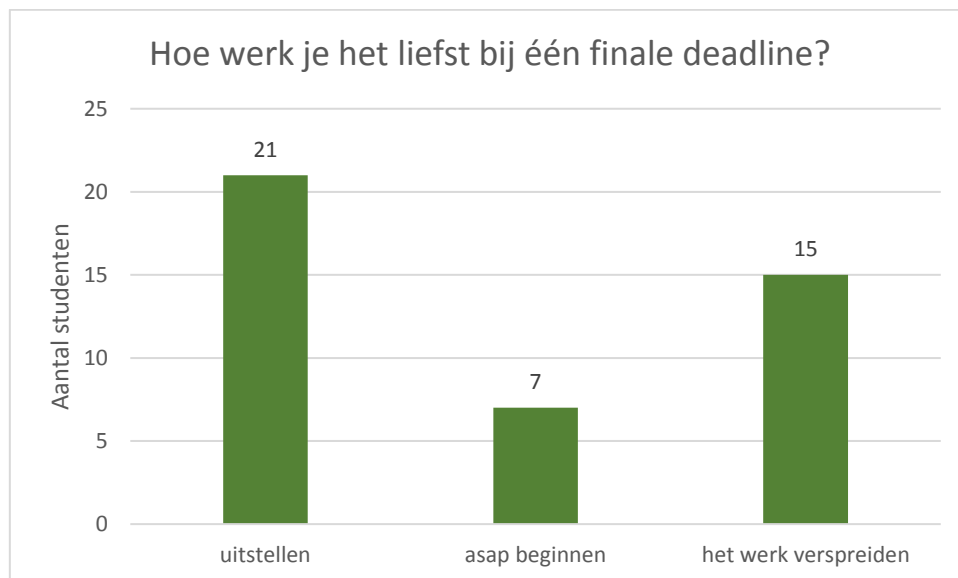
doorvoreren. Dit betekent dus dat de inzet niet alleen van de studenten maar ook van de docenten moet komen.

Principe 8: "Agile processen bevorderen constante ontwikkeling. De opdrachtgevers, ontwikkelaars en gebruikers moeten een constant tempo eeuwig kunnen volhouden."

Onderwijscontext: Uit de literatuurstudie, bij het onderdeel agile learning, werd er aangehaald door een student dat leren op een agile manier ervoor zorgt dat alles goed moet worden bijgehouden. In het hoger onderwijs is het zo dat er uitstelgedrag heerst en alles op het laatste moment wordt gedaan. Er moet dus worden voor gezorgd dat de studenten bij het maken van een project vanaf dag 1 aandachtig en gemotiveerd zijn zodat er geen achterstand opgelopen kan worden en dat ze dit tempo kunnen aanhouden.

Bij principe 1 werd aan de studenten in de inleidende enquête de vraag gesteld of ze het liefst met tussentijdse deadlines werken of met één finale deadline. De vraag die hierop volgde was hoe het liefst gewerkt wordt wanneer er één finale deadline is. Zoals te zien is op figuur 26 is het opvallend hoeveel studenten uitstellen bij één finale deadline (50%). De redenen die hiervoor werden gegeven zijn:

- de studenten stellen in het algemeen graag uit, en;
- er zijn andere deadlines gedurende het semester waar ook aan gewerkt moet worden.



Figuur 26: De manier van werken bij één finale deadline

Voor de respondenten die het werk liever verspreiden is de grootste reden dat het project tijdig klaar geraakt zodat er niet al te veel druk is op het einde wanneer alle deadlines samenkomen. Deze reden geldt ook voor diegene die voor onmiddellijk ('asap') beginnen hadden gekozen. Het toepassen van sprints van twee weken zorgt ervoor dat het uitstelgedrag teniet gaat en de studenten voortdurend met de zaak bezig zijn waardoor het leerproces ook verbeterd wordt. Als de hoeveelheid werk van een opleidingsonderdeel per week optimaal is afgesteld op de studiebelasting van een opleidingsonderdeel, is het normaal gezien mogelijk om tegen dit tempo constant te blijven werken. Dit wil niet zeggen dat de studenten elke dag met het opleidingsonderdeel bezig moeten zijn, maar

wel per week het gewenste resultaat afleveren en ze hierdoor geen achterstand oplopen. Er kan dus vanuit gegaan worden dat principe 8 toepasbaar is wanneer er gewerkt wordt met tussentijdse deadlines die ervoor zorgen dat de studenten verplicht zijn om zich regelmatig met het project bezig te houden.

Principe 9: "Voortdurende aandacht voor een hoge technische kwaliteit en voor een goed ontwerp versterken agility."

Dit principe is niet achterhaald door de enquêtes, aangezien het moeilijk was om een objectieve vraag op te stellen die nagaat in welke mate de studenten hun werk controleren op kwaliteit. Als dit principe vertaald wordt naar de hoger onderwijscontext, kan er aangehaald worden dat studenten ook voortdurend aandacht moeten hebben voor de kwaliteit van hun werk. Dit kan door het voltooide werk regelmatig te vergelijken met de kwaliteitscriteria die vooraf zijn vastgelegd (door henzelf, of door de docent), en door de feedback van de docenten. Er kan vanuit gegaan worden dat principe 9 toepasbaar is op het hoger onderwijs, mits de student aandacht heeft voor de kwaliteit van het product en het onderwijsteam regelmatig feedback geeft waardoor de kwaliteit wordt 'gecontroleerd'.

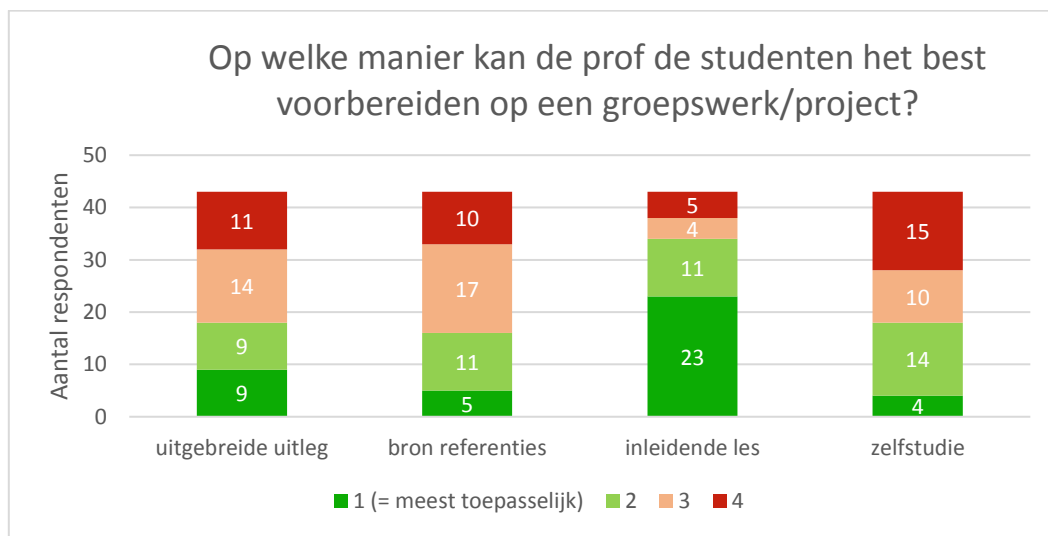
Principe 10: "Eenvoud, de kunst van het maximaliseren van het werk dat niet gedaan wordt, is essentieel."

Ook dit principe kon niet achterhaald door het afnemen van de enquêtes omdat dit slaat op het houden van een zo eenvoudig mogelijke softwarecode. Als dit principe vertaald wordt naar de hoger onderwijscontext, zou er kunnen aangehaald worden dat studenten hun werk zo concreet en beknopt mogelijk moeten uitvoeren. Dit kan doormiddel van een paginalimiet op te stellen van het af te leveren rapport of door de student erop te wijzen dat hij/zij de tekst zo concreet mogelijk moet houden. Hiermee wordt bedoeld dat er niet van de essentie moet afgeweken worden door te veel uitleg of herhaling over het behandelde onderwerp. Op deze manier leert de student hoe ze op een concrete manier goed werk kunnen leveren. Verder kan bijvoorbeeld bij Scrum de sprint retrospective van pas komen om na te gaan op welke manier het team te werk is gegaan en kan er nagedacht worden over wat er aan de werking van de groep verbeterd kan worden of hoe het werk beter of efficiënter kan gedaan worden. Er kan dus vanuit gegaan worden dat principe 10 toepasbaar is in het hoger onderwijs.

Principe 11: "De beste architecturen, eisen en ontwerpen komen voort uit zelfsturende teams."

Onderwijscontext: In een project in het hoger onderwijs is er vaak sprake van een leider die het voortouw neemt en ervoor zorgt dat het werk gedaan wordt waardoor het team in zekere mate niet zelfsturend is. Bij Scrum wordt er, zoals in de literatuurstudie werd vermeld, gewerkt met een ScrumMaster die het proces leidt en niet de mensen. De leider die zegt wat de teamleden moeten doen, zou moeten veranderen naar een leider die vertelt wat er moet gedaan worden, waarna de teamleden zelf bepalen wat ze doen en hoe ze dit doen. Verder is het zo dat wanneer er aan een project begonnen wordt, het team moet weten wat er van hen verwacht wordt en wat ze moeten kunnen. Zodra dit geweten is en de leider vergelijkbaar is met een ScrumMaster, kunnen ze zelfsturend worden.

Om de studenten voor te bereiden op een projectwerk, is er in de inleidende enquête aan de studenten gevraagd om een ranking te maken over de methode die zij verkiezen. Zo is in figuur 27 te zien dat er een voorkeur is voor een inleidende les met uitgebreide informatie over het onderwerp.



Figuur 27: Een ranking over de voorkeur omtrent het voorbereiden op een project

Zo is er ook te zien dat enkele studenten verkiezen dat er een uitgebreide uitleg van het groepswerk wordt gegeven met de methodes van uitwerking die er gebruikt moeten worden. Verder zijn er ook studenten die referenties naar bronnen met extra informatie verkiezen en studenten die liever zelfstudie doen omtrent het onderwerp en hier een vragenmoment bij het onderwijsteam krijgen in de week na opzet van het project. Dit toont aan dat er studenten zijn die willen dat er verteld wordt wat er moet gebeuren en hoe dit moet gebeuren, maar dat er ook studenten zijn die op zelfstandige basiskennis omtrent het onderwerp van het project willen vergaren. Aangezien een meerderheid voor de inleidende les op de eerste plaats koos, is een inleidende les met de uitleg van het project aangeraden. Hier kan ook een onderscheid gemaakt worden tussen het niveau van de studenten. Zo zullen de studenten van de 1^{ste} bachelor meer nood hebben aan een inleidende les omdat ze het zelfstandig zoeken naar informatie nog niet onder de knie hebben. In tegenstelling tot de 1^{ste} bachelor student, heeft een master student wel de nodige bagage om zelfstandig opzoekwerk te doen en zou deze geen nood hebben aan een inleidende les. Zo kunnen er extra bronnen beschikbaar gesteld worden zodat de master studenten ook op zelfstandige basis uitgebreid kunnen onderzoeken op welke manier ze het best werken. Wat wel zo is, en al is aangehaald aan het begin van dit hoofdstuk, is dat de studenten de juiste interpretatie van de verschillende elementen van een agile methode hebben zodat ze geen foute groepsbeslissingen nemen. Dit kan vermeden worden door een inleidende les met uitleg over de meest relevante en 'moeilijk' te interpreteren elementen.

Vervolgens werd er in de inleidende enquête ook gevraagd of de respondenten denken dat een groepswerk tot een goed einde gebracht kan worden zonder een leider die het team stuurt. 27 studenten waren van mening dat dit wel mogelijk is. De meest voorkomende redenen hiervoor waren:

- "Als alle teamleden initiatief en hun verantwoordelijkheden nemen en weten wat ze moeten doen is dit mogelijk"

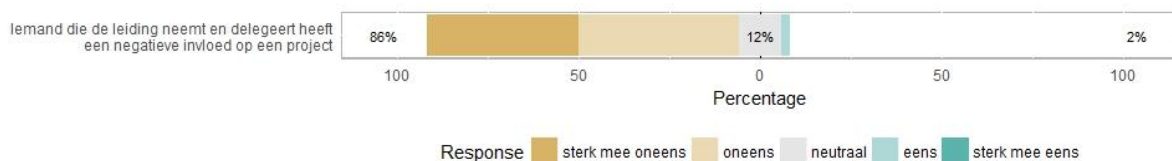
- "Elk lid moet actief participeren en er moeten in het begin goede afspraken gemaakt worden in verband met communicatie"

Er zijn een paar opmerkingen die wel interessant zijn vanuit het agile perspectief:

- "Niet 1 leider maar verschillende mensen die afwisselend het project coördineren"
- "Als er een goede sfeer heerst in de groep en als de groepsleden elkaar goed kennen is er niet echt een leider nodig, maar wanneer de groepsleden elkaar niet kennen is het volgens mij wel nodig"

Er zijn 16 studenten die beweren dat zonder een leider een groepswerk niet tot een goed einde kan gebracht worden. De meest voorkomende opmerking hier was dat er een leider moet zijn om de controle over het project te behouden zodat er bijvoorbeeld niet wordt uitgesteld. Een andere opmerking was dat er een leider nodig is om een goede taakverdeling op te maken zodat er geen dubbelwerk gedaan wordt. Deze opmerkingen hangen nauw samen met het soort leider dat een agile project nodig heeft.

In de inleidende enquête werd gevraagd welke factoren een negatieve invloed hebben op het maken en voltooien van een project. Iemand die de leiding neemt en delegeert was één van deze factoren. In figuur 28 is te zien dat 86 procent het oneens is met het feit dat een leidinggevend iemand een negatieve invloed heeft op een project. (42% sterk mee oneens, 44% oneens)



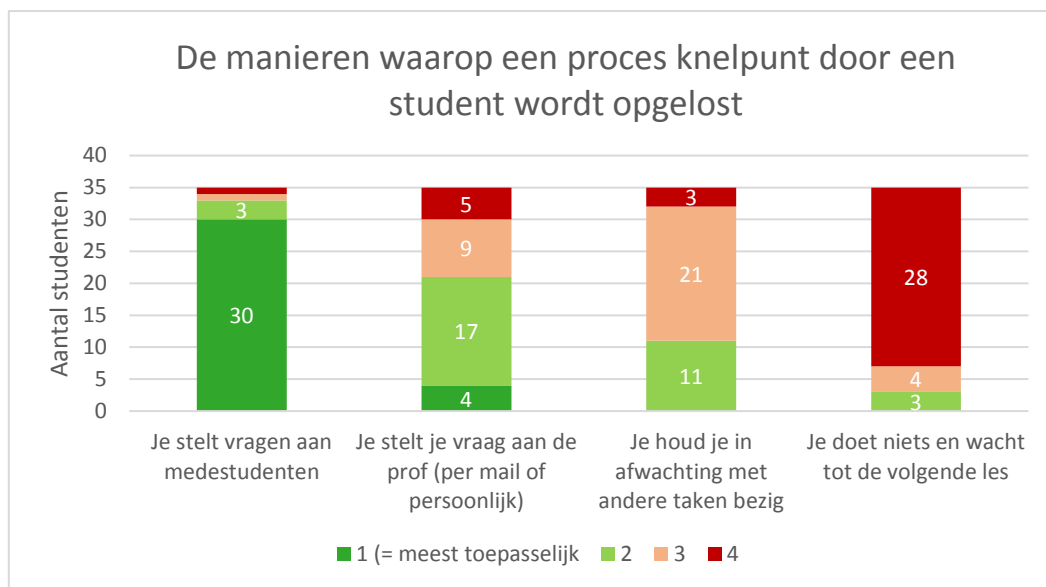
Figuur 28: De mate waarin een leider een negatieve invloed heeft op een project

In de tweede enquête werd ongeveer dezelfde stelling voorgelegd aan de studenten, maar op een tegenovergestelde manier. Zoals te zien is in figuur 29, was 77% van de respondenten het er mee eens of sterk mee eens dat een leider die zegt wat de teamleden moeten doen een positieve invloed heeft op een project. Als dit vergeleken wordt met dezelfde vraag uit de inleidende enquête, kan er vastgesteld worden dat er een daling was van 21,8 procent punt van de mensen die het sterk mee eens zijn dat een leider een positieve invloed heeft op een project. Dit zorgde er wel voor dat er een stijging was van 12,9 procent punt bij diegene die het er mee eens zijn. Verder was Neutraal toegenomen van 11,02% naar 20%. Deze veranderingen tonen aan dat het mogelijk is dat de voorkeur voor een leider voor een deel is afgenomen door het gebruik van Scrum en dat de voorkeur lichtjes gewijzigd is richting zelfsturende teams. Maar de verschillen tussen beide resultaten zijn niet groot genoeg om hier een gegronde uitspraak over te doen.



Figuur 29: De mate waarin een leider een positieve invloed heeft op een project

Aangezien er verwacht wordt dat de teams zelfsturend zijn, werd er in de tweede enquête gevraagd om te rangschikken welke probleemaanpak er gebruikt werd wanneer de studenten vast kwamen te zitten gedurende het proces. Het is duidelijk uit de resultaten in figuur 30 dat de studenten een voorkeur hebben voor het stellen van de vraag aan de medestudenten gevolgd door het stellen van de vraag aan het onderwijsteam. In een Scrumteam gaan de teamleden elkaar helpen de problemen op te lossen alvorens ze dat aan anderen vragen. Als het dan nog niet duidelijk is kunnen ze met hun vragen terecht bij de klant (vragen over de manier van werking, niet het technische aspect). Deze resultaten tonen ook aan dat deze studenten al een goede stap richting zelfsturing hebben gezet, aangezien ze onderling problemen proberen op te lossen en als dit niet lukt, ze zich wenden tot het onderwijsteam.



Figuur 30: Een ranking over manieren om een proces knelpunt op te lossen

Tot slot werd er in de inleidende enquête een vraag gesteld die naging of er volgens de studenten een onderscheid moet zijn tussen projecten in de bachelor en projecten in de master. 28 respondenten waren van mening dat er een onderscheid moet zijn en de andere 15 respondenten vonden van niet. Bij diegene die vonden dat er een onderscheid moet zijn, waren enkele redenen:

- "Meer begeleiding in de bachelor en in de master meer op een zelfstandige manier kunnen werken."
- "Doordat (academische) bachelor meer theoretisch is en master meer praktisch gericht is. "
- "Communicatie in master mag een stuk formeler en professioneler"
- "Projecten in master zijn minder begeleid. Dit verhoogt wel de moeilijkheid, maar stimuleert ook het zelfstandig werken. "

Door de respondenten die vonden dat er geen onderscheid moet zijn, werden geen inzichtvolle opmerkingen gegeven.

Aan de hand van de voorgaande resultaten, kan er worden gesteld dat principe 11 toepasbaar is, mits een aantal voorwaarden. Eerst en vooral moet er duidelijk vermeld worden wat er verwacht wordt van de studenten. Daarnaast moet er een leider in het team zijn die de functie van een

ScrumMaster uitoefent. En ten slotte moeten de procesproblemen die zich voordoen gedurende het project zo veel mogelijk door de studenten zelf worden opgelost.

Principe 12: "Op vaste tijden, onderzoekt het team hoe het effectiever kan worden en past vervolgens zijn gedrag daar op aan."

Onderwijscontext: Dit principe kan in het hoger onderwijs, wanneer er volgens een agile methode gewerkt wordt, ook volledig overgenomen worden.

Zoals aangehaald in de literatuurstudie wordt er bij Scrum gebruik gemaakt van de sprint retrospective waarbij er na een sprint gekeken wordt naar wat er beter kan. Ook bij de daily Scrum wordt er nagedacht over de problemen die er zich de vorige dag hebben voorgedaan. Uit de resultaten die bij principe 1 werden vermeld over de elementen/handelingen van Scrum die door de studenten werden toegepast, ontbrak de sprint retrospective in 13 van de 35 gevallen. Dit toont aan dat hier geen nadruk op is gelegd door het onderwijsteam. Verder kan zelf- en groepsreflectie ook helpen bij het onderzoeken hoe het team effectiever kan werken. Dit kan bijvoorbeeld gedaan worden door middel van een (tussentijdse) peerevaluatie zodat de groepsproblemen gedetecteerd kunnen worden.

Er kan geconcludeerd worden dat principe 12 toepasbaar is, mits de sprint retrospective door het onderwijsteam benadrukt wordt, er gebruik gemaakt wordt van een 'daily' Scrum-meeting op de vaste contactmomenten en er gebruik gemaakt wordt van een of meerdere peerevaluatiemomenten.

3.4 Beperkingen van deze studie en verder onderzoek

De enquêtes die werden afgenomen in deze studie, zijn ingevuld door studenten van de opleiding beleidsinformatica en informatica. Aangezien de meerderheid van deze studenten theoretische voorkennis had van agile projectmanagement, kan dit ervoor gezorgd hebben dat deze studenten bevooroordeeld waren, waardoor ze de vragen van de enquête hebben ingevuld met de theorie achter agile in het achterhoofd. Verder kan de grootte van de steekproef en het feit dat het geen random sample is, invloed hebben op de veralgemeenbaarheid van de resultaten.

In verdere studies zou deze enquêtes afgenomen kunnen worden bij een breder doelpubliek. Deze enquêtes zouden bijvoorbeeld kunnen afgenomen worden bij andere opleidingen (die eventueel een implementatie van agile methodes kent in een opleidingsonderdeel). Hiermee wordt bedoeld dat de sample vergroot kan worden om na te gaan of de elementen die bij de kleine sample zichtbaar waren ook zichtbaar zijn bij de grotere sample zodat er beter veralgemeend kan worden.

3.5 Conclusie

Bij het onderzoeken naar de toepasbaarheid van de twaalf leidende agile principes in het hoger onderwijs, is er een case onderzocht doormiddel van enquêtes af te nemen bij studenten. Zo werd er in 3.2 achterhaald wat de voorkennis van agile projectmanagement was bij de studenten en hoe ze de werking van het project, waarin ze Scrum moesten toepassen, ervaarden. De meerderheid van de studenten hadden voorkennis van bepaalde methodes en de werking van het project verliep vlot bij praktisch iedereen. Verder werden in 3.3 de twaalf principes omgevormd naar de hoger onderwijscontext en werden deze principes onderbouwd met resultaten uit de enquêtes. Zo is het

mogelijk om een antwoord te formuleren op de eerste deelvraag van deze thesis, namelijk: "Welke principes van agile projectmanagement zijn toepasbaar in het hoger onderwijs?". Er is tot de vaststelling gekomen dat alle principes toepasbaar zijn in het hoger onderwijs, mits er bepaalde aanpassingen worden doorgevoerd zodat de focus niet ligt op het softwareontwikkelingsaspect. Aangezien de vragen zijn gesteld op basis van bevindingen uit de literatuurstudie en er gebruik gemaakt werd van een theoretische sample, kan ervan uitgaan worden dat deze bevindingen (analytisch) veralgemeenbaar zijn. Om deze bevindingen te versterken en aan te vullen, werd er in het volgende hoofdstuk een observatie uitgevoerd die de verschillen tussen de hoger onderwijscontext en bedrijfscontext schetst, alsook gaat nagaan of bedrijven zich volledig houden aan de twaalf leidende principes, met andere woorden, zich houden aan het basis artefact van deze case study. Tot slot zullen de aanpassingen of voorstellen die in dit hoofdstuk werden aangehaald, door zowel het onderwijsteam als de studenten, moeten doorgevoerd worden om deze principes te laten gelden. Deze aanpassingen zullen in de vorm van richtlijnen in een tabel voorgesteld worden in hoofdstuk 5.

Hoofdstuk 4: Agile projectmanagement in de bedrijfscontext

4.1 Inleiding

Zoals aangehaald bij de probleemstelling en in de literatuurstudie, is agile projectmanagement ontworpen om in softwareontwikkelingsbedrijven toe te passen. De 12 principes en de kernwaarden zouden door bedrijven gehanteerd moeten worden, alsook de elementen die werden aangehaald in de literatuurstudie bij sectie 2.5.

Om na te gaan of een bedrijf van deze richtlijnen ook werkelijk gebruik maakt, is er een meeloopdag georganiseerd in een bedrijf dat zichzelf ziet als een agile bedrijf. Dit bedrijf wordt in het kort geschetst in sectie 4.2. Dit stelt de tweede case voor en om deze case te onderzoeken werd er in het bedrijf geobserveerd hoe de werknemers te werk gingen. De werknemers worden hier ook weer als een theoretische sample beschouwd aangezien zij volgens verschillende agile methodes werken en dus de twaalf leidende principes in acht moeten nemen. Verder zijn er ook vragen gesteld aan bepaalde werknemers om meer informatie omtrent de werking van het bedrijf te vergaren. Al deze elementen en handelingen zijn terug te vinden in sectie 4.3. Bij deze meeloopdag was er een professionele agile coach aanwezig die een simulatie deed samen met de werknemers. Het doel van deze simulatie was om de werknemers in te laten zien waar de problemen in de flow van het bedrijf zich kunnen voordoen en hoe deze kunnen worden opgelost. Na deze simulatie heeft de agile coach ervaringen gedeeld omtrent agilititeit. De inbreng van deze agile coach is terug te vinden in sectie 4.4. Verder zijn de beperkingen van deze studie aangehaald in sectie 4.5.

Uit de observatie, de gegenereerde input van de werknemers en de ervaringen van de agile coach, kon er een vergelijking gemaakt worden met de onderwijscontext uit eigen ervaring en uit de bevindingen van hoofdstuk 3, en kan een antwoord op de tweede deelvraag geformuleerd worden. Deze vergelijkingen zijn terug te vinden in sectie 4.6. De bevindingen zullen ook steeds weer gekoppeld worden aan de principes zodat deze bepaalde richtlijnen die in het vorige hoofdstuk werden beschreven kunnen versterken en aanvullen. Omwille van confidentiële redenen zullen de naam van het bedrijf, de naam van de werknemers en de naam van de agile coach niet vermeld worden in het verdere verloop van deze thesis. Het bedrijf zal voorgesteld worden als bedrijf X.

4.2 Het agile bedrijf

Het bedrijf waar deze meeloopdag is gebeurd, is een bedrijf dat al 15 jaar innovatieve HR-software ontwikkelt, loon- en personeelsadministratie doet en sociaal juridisch advies geeft. Het bedrijf telt meer dan 100 medewerkers. Ze streven ernaar om zowel huidige als toekomstige interimkantoren, dienstenchequebedrijven en ondernemingen met elke HR-gerelateerde vraag verder te helpen en zo de administratie te beperken.

4.3 De werking en elementen van het bedrijf vergeleken met het hoger onderwijs

Om de vergelijking te kunnen maken met het onderwijs, zullen hierna de verschillende elementen die zichtbaar waren in het bedrijf of verteld werden, aangehaald worden en vergeleken worden met het hoger onderwijs.

4.3.1 Indeling van de werkvloer

Het viel op dat de indeling van de hele ruimte min of meer op een agile manier is. In het midden bevindt zich een tafel waar iedereen kan zitten met het algemeen bord aan de ene kant en aan de andere kant een keuken waar de werknemers koffie kunnen benuttigen alsook hun eten opwarmen. De eilandjes zijn allemaal rondom dit centrum gelegen en aan één zijde bevonden zich ook enkele vergaderruimtes, allemaal open en transparant. De indeling van de werkvloer werd aangehaald in de literatuurstudie bij het deel over extreme programming. Hier werd er vermeld dat er rondom het centrum allemaal cabines zijn en dat in het midden computers gesitueerd staan waar de pair programmers samen kunnen werken. Op deze manier kan de communicatie en samenwerking tussen iedereen optimaal verlopen.

Een indeling zoals in dit bedrijf, is moeilijk te implementeren in het hoger onderwijs aangezien de gebouwen in een hogeschool of universiteit voornamelijk bestaan uit leslokalen waar er zich alleen tafels, stoelen en een schoolbord bevinden. Aangezien de leslokalen in de meeste gevallen door verschillende klassen en opleidingen worden gebruikt, is het moeilijk deze aan te passen naar een agile omgeving voor één of enkele specifiek(e) project(en). Wat hier moet aangehaald worden is dat in een hoger onderwijscontext de verschillende groepen in de meeste gevallen niet met elkaar samenwerken. Hierdoor is het niet nodig om de indeling van de werkomgeving identiek te maken als dat van het bedrijf. Het maken van werkeilandjes met verschillende tafels is voldoende. Wat ook is aangeraden, dit hangt ook samen met principe 5 (ondersteunende omgeving), is om de studenten van een projectruimte te voorzien waar ze samen kunnen werken.

4.3.2 Visueel management bord per departement

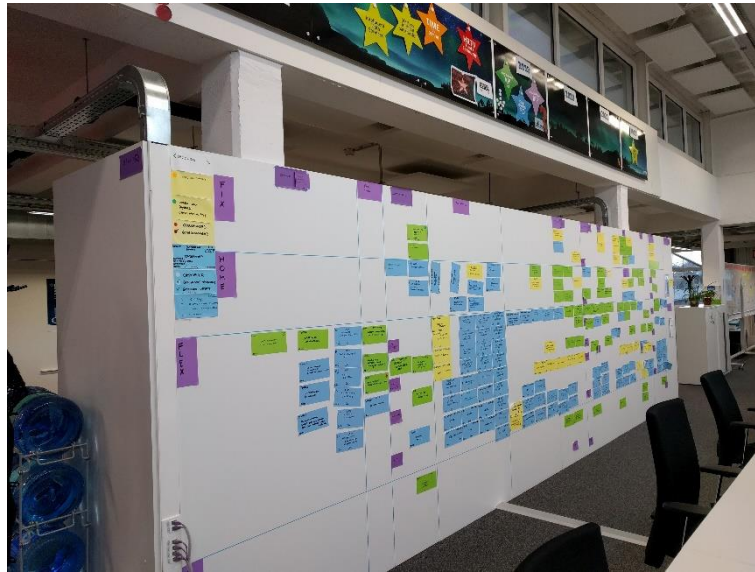
Bij het visualiseren van de werkvloer was ook meteen op te merken dat de ruimte ingedeeld is in verschillende departementen (werkeilanden). Elk departement heeft een eigen visueel management bord waarop de onderdelen van een story of meerdere stories verspreid hangen in de fase waarin het zich bevindt. Zo kan er gezien worden wie met welke taak bezig is, welke taken belangrijk zijn, welke taken er nog moeten gedaan worden en waar er zich problemen bevinden (bijvoorbeeld: veel geblokkeerd werk in een bepaalde stap of een team dat overbelast is of net niet genoeg werk kan binnenhalen). Deze fysieke borden dienen ook ter coördinatie met externe teams.

Soms worden ook 'avatars' gebruikt, een magneet met het gezicht van de werknemer, op het kaartje van de taak om te zien wie er met welke taak bezig is. Deze borden worden geregeld aangepast. Verder was er ook een globaal bord in het midden van de werkvloer gesitueerd, zoals te zien is op figuur 31.

Dit algemeen bord schetst het hele verhaal (maar zonder al de details die wel aanwezig zijn op niveau van de teams) en toont de flow van de processen die er zich afspelen binnen het bedrijf. Op dit algemeen bord zijn er drie soorten kaartjes. De blauwe kaartjes stellen het 'billable' werk voor wat bestaat uit zowel 'standaard' werk (kant en klaar, met weinig onzekerheid) maar ook 'maatwerk' (met grotere onzekerheid). Verder zijn er groene kaartjes die gebruikt worden voor IP (intellectual property) creation, wat investering in productinnovatie voorstelt. Tot slot zijn er gele kaartjes die 'maintenance' voorstellen en die bij de interne visuele management borden bij de departementen ook vaak fouten voorstellen die gecorrigeerd moeten worden. "Hoe meer blauw, hoe beter!" luidde

de boodschap! Het is echter de balans tussen de soorten werk die op korte termijn en lange termijn het bedrijf gezond houden.

Het is praktisch onmogelijk om voor elk opleidingsonderdeel waar een project wordt gedaan, een tastbaar visueel management bord voor elk team te voorzien op een campus. Een oplossing hiervoor is, zoals eerder vermeld, het gebruik van online tools zoals Trello, waar een visueel management bord online gemaakt en gedeeld kan worden. Zo een tool of een tastbaar visueel management bord is aan te raden wanneer een project volgens de agile methodologie gedaan wordt. Dit kan gelinkt worden aan principe 6 en werd ook aangehaald in hoofdstuk 2 en 3.



Figuur 31: Algemeen visueel management bord

Bij het front-office departement werd er door een werknemer aangehaald dat ze hun bord een half jaar geleden gedigitaliseerd hebben. De reden hiervoor is dat zij met Scrum werken en dat ze op deze manier burndown charts kunnen maken waardoor ze hun vordering zien. Een andere reden voor de digitalisering is dat er twee teams zijn die aan hetzelfde product werken en dat het voor hun vooral nodig was om makkelijker te kunnen tracken welke features in een bepaalde release zitten. Het wordt dus gebruikt voor het verbeteren van interne communicatie, waardoor er geen dubbelwerk wordt gedaan, waardoor er geweten is wie met wat bezig is en hoe ver ze staan met het voltooien van een bepaald feature. Wanneer dit wordt toegepast in een onderwijscontext, zullen alle studenten van deze voordelen ook kunnen genieten.

4.3.3 Het gebruik van agile methodes

Zoals aangehaald werd in de literatuur, is er tegenwoordig meer en meer gebruik van hybride methodes die zowel verschillende agile methodes met elkaar combineren als agile met traditionele methodes combineert. Bedrijf X maakt geen gebruik van één specifieke agile methode, maar gebruikt een mix van methodes (bijvoorbeeld: lean, kanban, Scrum, XP...). Ze werken per departement met zelf-gedisciplineerde gemotiveerde cross-functionele teams. Deze teams bestaan voornamelijk uit ontwikkelaars en elk team heeft ook een analist. Deze analist valt te vergelijken met de product owner bij Scrum die de backlog beheert en ervoor zorgt dat de juiste delen, het functionele 'WAT' waaraan gewerkt wordt in het team, van de backlog in de desbetreffende sprint worden voltooid.

Hij/zij zal ook alle afgewerkte deeltjes testen en valideren of het voldoet aan de noden van de klant. De analist is de persoon die het dichtst bij de klant staat en wordt ook beschouwd als een 'proxy' klant. Als er vragen of onzekerheden zijn bij de teamleden gaan zij eerst de analist hierover aanspreken. Als de analist deze vragen niet kan beantwoorden gaat deze de klant contacteren om dit probleem op te lossen. De developers bepalen dan 'HOE' dat work item gebouwd wordt.

Wanneer er een project gedaan wordt in het hoger onderwijs, is het niet vanzelfsprekend dat er gebruik gemaakt wordt van een mix van verschillende methodes. Als de studenten geen kennis hebben van een agile methode, is het aangeraden om het project volgens één agile methode aan te pakken. Dit kan in combinatie met elementen van traditionele methodes, maar aangezien de bedoeling is dat de studenten op een agile manier leren werken, is het beter dat ze enkel de aangeleerde eventueel aangepaste (naar het hoger onderwijs) methode toepassen.

De docent wordt in het hoger onderwijs beschouwd als de product owner en als klant wanneer er volgens een agile methode gewerkt wordt. Deze bepaalt de opdracht die er uitgevoerd moet worden en in veel gevallen ook de deelopdrachten en zal deze ook valideren. De docent moet de studenten sturen om deze deeltaken zo goed mogelijk uit te voeren alsook open staan voor vragen van de studenten wanneer er problemen zijn. Zo wordt de relatie tussen de 'product owner'/'klant' en het team bevorderd. Het is ook aan de docent om de agile methode die er gebruikt wordt op een juiste manier aan de studenten aan te leren, zoals werd aangehaald in hoofdstuk 3 waar er verteld werd dat een inleidende les ervoor kan zorgen dat de studenten een juiste interpretatie hebben van de verschillende elementen.

Verder werd er opgemerkt door de agile coach dat werken op een agile manier niet persé betekent dat er een bepaalde methode gevolgd wordt. Hij benadrukte het feit dat de principes gevolgd moeten worden zodat dit voor elk type project gebruikt kan worden.

4.3.4 Samenwerking met de klant

Bij de agile methodologie wordt ervan uitgegaan dat er zoveel mogelijk (lieft dagelijks) met de klant wordt samengezeten om de veranderende noden te identificeren en hierop te reageren. Bij bedrijf X wordt er zoveel mogelijk met de klant gecommuniceerd, maar dit gebeurt niet op een dagelijkse basis aangezien zij een groot aantal klanten hebben voor 'hetzelfde' type product. Het bedrijf werkt met cadence (vergelijkbaar met sprints bij Scrum) van twee weken die steeds wordt afgesloten met een demo. Deze demo is meestal in de vorm van een voorstelling waarbij de werking van een stuk (belangrijke) code wordt getoond. Na twee cycli, dit betekent na één maand, is er een product release waarop de klant feedback kan geven op een deel gereed product in de vorm van software. Dit kan gelinkt worden aan principe 1 waarbij er vroegtijdig en voortdurend waardevolle software aan de klant geleverd wordt. Verder kan principe 4 hier ook aangehaald worden aangezien hier vermeld wordt dat mensen uit business en ontwikkelaars dagelijks moeten samenwerken. Dit principe wordt deels (langs het perspectief van de klant) ontkracht aangezien er geen nood is om dagelijks te communiceren met de klant. Dit principe is in hoofdstuk 3 ook aangepast naar 'op regelmatige basis' samenwerken.

Dit toont aan dat wanneer er volgens een agile methode gewerkt wordt bij een project in het hoger onderwijs, er geen noodzaak is om dagelijks met het onderwijsteam samen te werken aangezien het

team op zelfstandige basis verder kan werken tot het volgende contactmoment. Wanneer de werklast is afgesteld op de te voltooien sprints, zijn deze contactmomenten voldoende om vragen te stellen aan het onderwijsteam.

4.3.5 Werken aan een constant tempo

In een bedrijf is het vanzelfsprekend dat er continu gewerkt moet worden aangezien de werknemers betaald worden om te werken en er verwacht wordt dat er aan een constant tempo gewerkt wordt. Om te achterhalen of dit ook echt zo is, werd er een aan een werknemer gevraagd of er soms sprake is van uitstelgedrag. Hij antwoordde dat door het werken met twee sprints per product release het soms wel opvalt dat de vooruitgang in de eerste sprint minder groot is ten opzichte van deze in de tweede sprint. Verder was te zien dat de burndown chart soms ook opwaarts gaat zoals in de literatuur werd vermeld. Ook hier is de reden dat ze met twee sprints werken voor de product release waardoor er een kans is dat er in de tweede sprint nog werk bij komt. Er is dus wel sprake van uitstelgedrag, maar aangezien ze op een agile manier werken is dit beperkt. De agile coach vertelde hierbij dat er hier gezien kan worden dat het team de Scrum rituelen gebruikt om te coördineren, maar dat het werk eerder stroomt met een te verwachte lead time van minder dan 1 maand. Dit kan gekoppeld worden aan principe 8 wat duidt op het feit dat agile processen een constante ontwikkeling bevorderen.

In een hoger onderwijscontext zal dit soort uitstelgedrag ook zeker zichtbaar zijn, zoals in hoofdstuk 3 uit de resultaten bleek. Wanneer er gewerkt wordt met sprints van bijvoorbeeld twee weken, zullen de studenten de neiging hebben om deels uit te stellen, of goed te starten gevolgd door uitstelgedrag. Dit werd aangehaald door de studenten bij de vraag over het uitstelgedrag bij één finale deadline in de inleidende enquête. Hier was er een meerderheid die uitstelt omdat dit gewoon kan en er andere deadlines zijn die 'voorrang' krijgen. Wel is het zo dat wanneer er op een agile manier en met sprints van twee weken gewerkt wordt, dit uitstelgedrag grotendeels teniet gaat door de kortere periodes waarin iets klaar moet geraken.

4.3.6 Communicatie voor dagelijkse coördinatie

Op vlak van communicatie voor dagelijkse coördinatie tussen de teamleden en departementen werkt bedrijf X met standups (vergelijkbaar met de daily Scrum meeting in Scrum). Elke ochtend om negen uur doet elk team een standup meeting van ongeveer 15 minuten. Sommige teams volgen het Scrum formaat voor deze meeting; wat er de dag ervoor is gebeurd, wat er die dag moet gebeuren en wat de belemmeringen zijn die zouden moeten worden opgelost. In essentie focust de standup van de teams op belemmeringen identificeren en verzekeren dat dit (samen) wordt opgelost en kijken wat kan opgeleverd worden in de geest van; "stop starting, start finishing". Na de standup meeting is er om kwart na negen een bijkomende standup meeting waarbij van elk departement enkele personen samenkomen om de overkoepelingen tussen de departementen te bespreken. Op deze manier weet iedereen precies wat er moet gebeuren en zal de flow zo optimaal mogelijk verlopen.

Als er doorheen de werkdag nog vragen zijn kan er altijd via de interne telefoon gecommuniceerd worden en als het heel dringend is kan er naar een departementseiland gelopen worden om de vraag face-to-face te stellen. Verder wordt er, zoals eerder vermeld, voor de dagelijkse coördinatie gebruik gemaakt van de visuele management borden. Tot slot zijn er sommige teams binnen het bedrijf ook

deels overkoepelend of werken ze samen met teams op een andere locatie (via constante video conference). Dit kan gelinkt worden aan principe 4 en 6. In dit bedrijf werken de mensen dagelijks samen en aangezien ze werken met standups, is er sprake van voldoende face-to-face communicatie waardoor de communicatie zo efficiënt en effectief mogelijk verloopt.

In een hoger onderwijscontext is het onmogelijk, zoals vermeld werd in hoofdstuk 3, dat de studenten dagelijks met elkaar afspreken om aan het project te werken. Hierdoor is face-to-face communicatie ook niet altijd mogelijk is. Dit toont aan dat er op vlak van communicatie een groot verschil is tussen de bedrijfscontext en hoger onderwijscontext. Aangezien er werd aangetoond in hoofdstuk 3 dat dagelijks samenwerken niet nodig is, maar er wel voldoende contactmomenten moeten zijn en een goed communicatiekanaal gebruikt wordt, vormt dit niet bepaald een probleem.

4.3.7 Collaboratie en interactie

Aangezien het bedrijf een softwareontwikkelingsbedrijf is en er gewerkt wordt met verschillende teams is collaboratie en interactie binnen het bedrijf ook heel belangrijk. Zo is er vaak te zien dat een werknemer van een departement samen langs een andere werknemer van hetzelfde departement zit om deze te helpen. Het bedrijf werkt met een 80-20 regel waarbij 80% van al het werk gaat naar het voltooien van de taken die de werknemers zelf opgelegd worden en 20% naar het helpen van collega's en eventueel het uitvinden van eigen stories voor in de product backlog. Het bedrijf doet ook aan pair programming, maar op een beperkte manier. Ze gebruiken pair programming meestal aan het begin van een story en gaan daarna individueel verder. Een werknemer gaf als opmerking dat het in zo een uitgebreide software heel moeilijk is om als individu alles te weten. Hiervoor is pair programming heel handig, want zo kan er kennis en kunnen inzichten uitgewisseld worden waardoor iedereen uiteindelijk op dezelfde golflengte zit. Ook dit kan weer gekoppeld worden aan principe 4 (regelmatig samenkomen) en 6 (voldoende face-to-face communicatie), maar ook aan principe 11 (zelfsturende teams) aangezien de werknemers elkaar helpen waar nodig en deze ook eigen initiatief nemen om bepaalde features uit te vinden.

Dit laatste werd ook vastgesteld uit de enquêtes die afgenomen werden bij de studenten. Bij een knelpunt van het proces gaan de studenten hulp vragen bij medestudenten alvorens ze naar het onderwijsteam gaan. Verder is het zo dat enkele studenten verkozen om samen in groep te werken, omdat dit tot meer inzichten leidde en dat het werk vlot verliep. Het is wel zo dat de studenten van mening waren dat de werklust te laag was om nog veel individueel werk thuis te doen. In het hoger onderwijs moeten studenten doen wat er van hen gevraagd wordt en alle extraatjes hebben voor de hoger onderwijscontext in principe geen meerwaarde. Natuurlijk kunnen de studenten deze extraatjes zien als individuele ontwikkeling wat in de toekomst een meerwaarde in het bedrijfsleven kan zijn.

4.3.8 Motivatie

Wat betreft de motivatie werd er gevraagd aan wat de werknemers motiveert om goed te presteren. Er werd vanuit gegaan dat het maandelijks loon (instrumentele waarde) dat de werknemers ontvangen de grootste motivator is. Dit was volgens enkele werknemers echter alleen de motivator gedurende de eerste werkmaanden. Er werd beweerd dat motivatie uiteindelijk komt vanuit het perspectief van het bedrijf. Het bedrijf probeert ervoor te zorgen dat de werknemers intrinsiek

gemotiveerd zijn. Dit doen ze door dankbaarheid te tonen voor de behaalde resultaten van de werknemers. Een individu is meestal ook intrinsiek gemotiveerd omdat hij/zij gespecialiseerd is in bepaalde zaken en hier beter in wil worden. Verder willen de werknemers ook meerwaarde toevoegen en een deel uitmaken van een groter geheel waardoor ze gemotiveerd worden (eindwaarde, want zorgt voor voldoening). Een uitdagend project (eindwaarde) zorgt natuurlijk ook voor extra motivatie, aangezien dit voldoening geeft wanneer het voltooid is. Tot slot voorziet het bedrijf beloningen (instrumentele waarde) zoals een vakantie wanneer ze verschillende mijlpalen bereikt hebben wat natuurlijk voor een bijkomende motivatie zorgt. Dit is verbonden aan principe 5 en laat ook weer zien dat de drie soorten waardes, die werden besproken in de literatuurstudie, in combinatie voorkomen. De werknemers hebben voldoende waarde, worden ondersteund door de omgeving (fysieke omgeving en ondersteuning van mede werknemers en baas) en hebben (door trainingen en opleidingen) het gevoel dat ze bekwaam zijn, wat alle drie samen zorgt voor motivatie.

In een hoger onderwijscontext werd er in de tweede enquête, gevraagd aan de studenten wat hun motiveert om goed te presteren. Hieruit bleek dat slagen de grootste motivator is, gevolgd door het uitbreiden van kennis en de uitdaging voor het maken van het project.

Het uitbreiden van kennis (intrinsieke waarde) en een uitdagend project (eindwaarde) kan als gelijkaardig beschouwd worden in een bedrijfscontext, maar de andere elementen verschillen grotendeels ten opzichte van de bedrijfscontext. Intrinsieke motivatie bij studenten om goed te studeren/te presteren zou bijvoorbeeld bereikt kunnen worden door real-life cases of voorbeelden te geven gedurende de hoorcolleges, maar ook door projecten/groepswerken te doen met real-life voorbeelden. Zo ontstaat er interesse in een bepaald topic en wordt deze materie beter verwerkt bij de studenten. Andere voorbeelden om studenten te motiveren werden weergegeven in sectie 2.8.5.

4.3.9 Overkoepelend 'Pacemaker' team

Een werknemer, de persoon die de rondleiding op de werkvloer heeft gegeven, vertelde dat hij in het Pacemaker team zit. Dit is het team dat zich focust op de ontwikkeling van Elvive. Elvive is de nieuwe loonmotor die bedrijf X in het leven heeft geroepen waardoor het bedrijf in een nieuwe value stream is terecht gekomen. Dit is een complex en grootschalig project (10.000 tot 20.000 werkdagen) waardoor het efficiënt organiseren van het elvive-werk parallel met al het andere werk een grote uitdaging is. Dit team probeert ervoor te zorgen dat de aanwezige kennis optimaal benut wordt en dat de bestaande teams productief meewerken zonder dat de nodige experts uit de bestaande teams van team zouden moeten veranderen.

Dit laat zien dat het bedrijf zo efficiënt mogelijk probeert te werken. Dit op een agile manier, want ze werken met kleine teams die intern met elkaar communiceren en samenwerken maar ook extern met de andere teams. Hierbij kan je vaststellen dat het bedrijf niet zozeer streeft naar het installeren van een vaste set agile praktijken maar eerder het agile denken probeert te stimuleren.

In het hoger onderwijs zullen er nooit projecten zijn van deze omvang en aangezien een project altijd door één team voltooid moet worden, is er in een hoger onderwijscontext geen nood aan een overkoepelend team. Wat wel moet worden aangehaald is dat het stimuleren van het agile denken bij de studenten bevorderd kan worden. Dit door hen ten eerste de theorie achter bepaalde agile methodes aan te leren, maar ook door hen te laten leren werken op een agile manier. Op deze manier

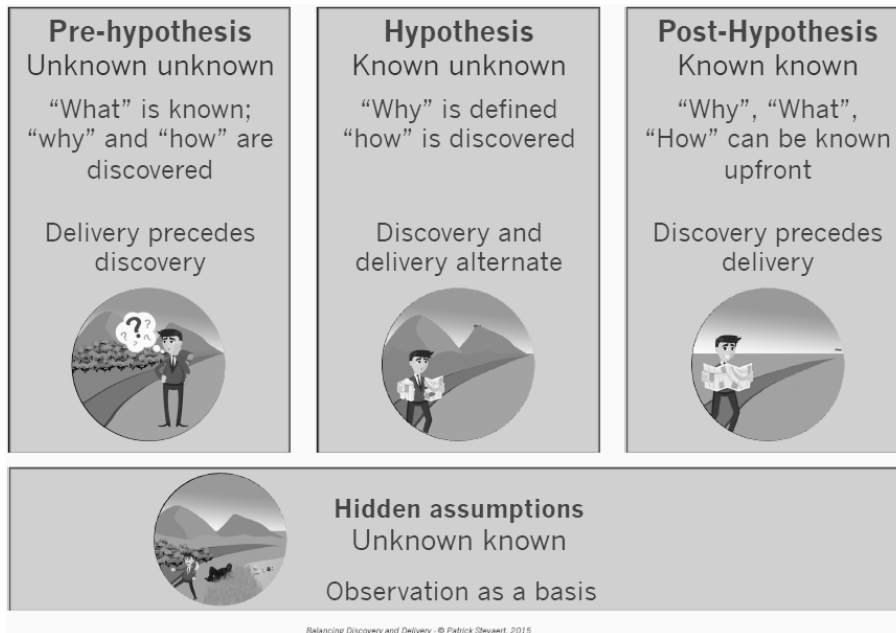
wordt dit een soort van automatisme en gaan ze dit automatisch toepassen bij andere projecten, waarbij er niet volgens een agile methode gewerkt wordt. Zo worden ze klaargestoomd voor het bedrijfsleven.

4.4 De inbreng van de agile coach

De agile coach die deze meeloopdag mogelijk gemaakt, heeft een eigen inbreng als aanvulling op de informatie vergaard op deze dag gegeven. Bij het behandelen van onzekerheid zei hij bijvoorbeeld dat de vraag die er zich (in de hoger onderwijscontext) gesteld moet worden is: "Waarom doen we dit project?", "Voor wie moeten we dit doen?", en "Is er een reden waarom we dit moeten doen?". Hij verwijst hier naar het feit dat 'wat' en 'hoe' al vaak geweten is maar 'waarom' en 'wie' meestal niet geweten is. Hij gebruikt hiervoor de uitspraak: "Delivery precedes discovery" wat valt onder de categorie "nurturing ugly ducklings". Dit laatste betekent dat je iets gaat maken maar het is onduidelijk of je wel een echt probleem van iemand of een groep mensen zal oplossen. Enkel door iets tastbaar aan te bieden gaan mensen kunnen ervaren of een 'job to be done' gemakkelijker wordt. Vaak vraagt dit tijd en doorzettingsvermogen. Vandaar moet de 'ugly duckling' gekoesterd en gevoed worden om een match te vinden tussen probleem en oplossing.

In een onderwijscontext is het vaak zo dat er geen reden wordt gegeven waarom iets wordt aangeleerd, zoals bijvoorbeeld integralen bij wiskunde. Als dit in een context wordt geplaatst (bijvoorbeeld: berekenen van raketbanen) waardoor studenten inzien waarvoor het werkelijk dient, kan er verondersteld worden dat (een deel van de) studenten gemotiveerder zullen zijn aangezien ze het nut ervan inzien. Het leerproces zal veel leerrijker zijn omdat er op deze manier ook beter onthouden zou kunnen worden. Hij verwees hiermee naar het feit dat er pas iets gekend is wanneer het verwerkt is, in het langetermijngeheugen zit en het kan gereproduceerd worden op verschillende niveaus (bijvoorbeeld: overzicht geven, detail uitleg geven, toepassen...). Dit kan gelinkt worden aan principe 5, want dit zorgt ervoor dat studenten gemotiveerder worden om goed te presteren. Hier werd ook aangehaald dat heel veel studenten uitstelgedrag vertonen en hierdoor alles in hun blokperiode moeten verwerken. Het probleem hier is dat studenten op een korte termijn teveel kennis proberen te vergaren waardoor ze het vaak maar tijdelijk 'kennen' (pagina's herkennen) en achteraf vergeten. In de veronderstelling dat de 'moeilijke vakken' en projecten met meerdere studenten ugly ducklings zijn dan is net tijd, regelmatige aandacht en doorzettingsvermogen over een langere periode de sleutel.

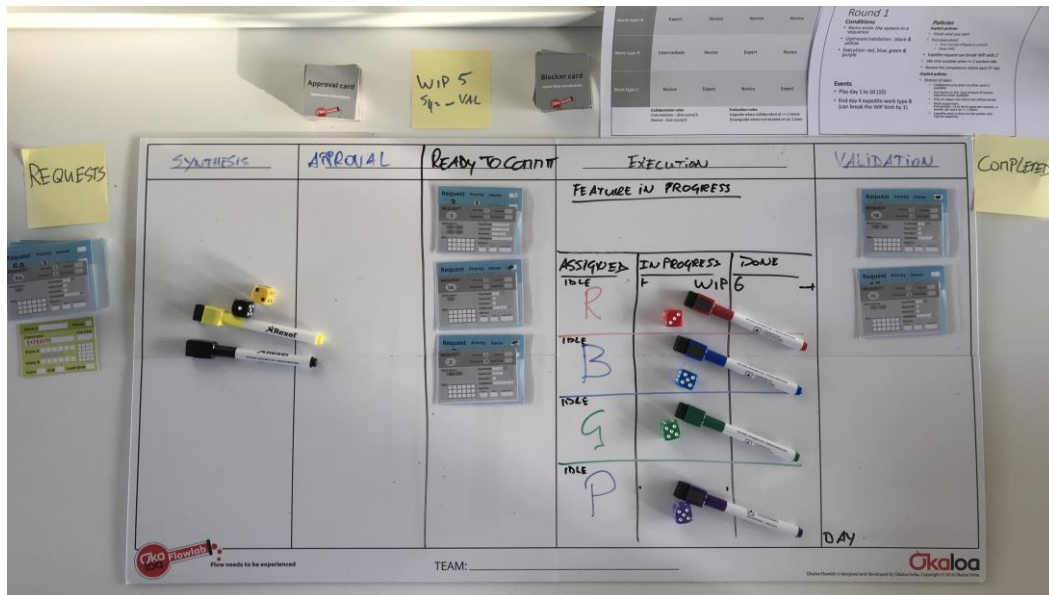
Bij agile development zal 'delivery' en 'discovery' afwisselen waarbij de 'waarom' en 'wie' bekend is en de 'wat' en 'hoe' nog ontdekt moet worden. Bij service delivery is het het omgekeerde als bij nurturing the ugly ducklings. Hier zal discovery vooraf gaan (bijvoorbeeld: een invulformulier) aan delivery waarmee bedoeld wordt dat 'waarom', 'wat', 'hoe', 'wie' en 'wanneer' bekend is. Deze drie verschillende manieren van het behandelen van onzekerheid worden voorgesteld in figuur 32.



Figuur 32: behandelen van onzekerheid (bron van de agile Coach)

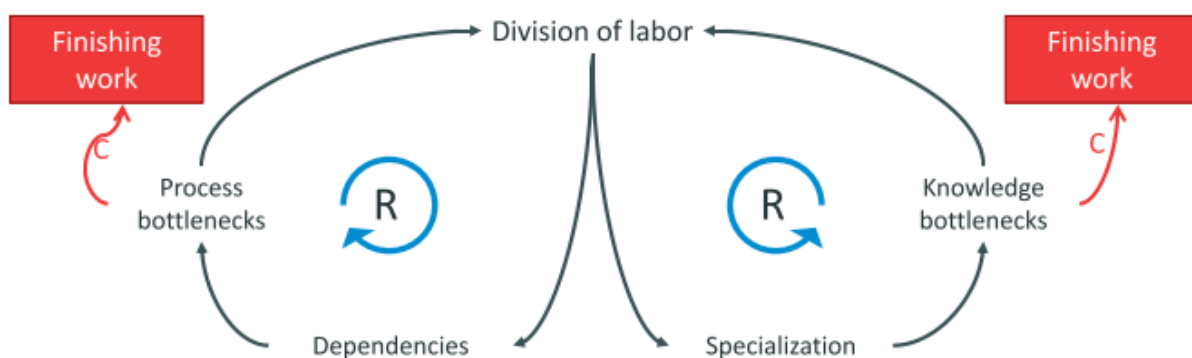
4.4.1 Okaloa flowlab simulatie (geleid door de agile coach)

Het was de tweede dag dat de agile coach in het bedrijf aanwezig was. De agile coach heeft gedurende deze dag, samen met vier andere werknemers van het pacemaker team, in een vergaderruimte de okaloa flowlab simulatie gedaan. Deze simulatie is ontworpen door deze agile coach samen met een andere specialist in het vak. Met deze vier werknemers had de agile coach de dag voordien een standaardsimulatie gedaan wat toonde dat samenwerken ervoor zorgt dat er meer werk gedaan kan worden en dat het werk zich op deze manier niet opstapelt. Deze simulatie was een uitbreiding op de andere en de focus hier lag eerder op het end-to-end proces. In dit simulatiemodel, zoals te zien is op figuur 33, wordt een onderscheid gemaakt tussen het deel dat zich bezighoudt met het aannemen van werk (de drie eerste kolommen links van de figuur) van de klant en het valideren van het resultaat (de meeste rechtse kolom). Ten tweede was er een deel (de 4 teams of teamleden, hiervan werd abstractie genomen en is terug te zien in het middelste deel op de figuur) dat zich bezighoudt met de uitwerking van de projecten. Zo een project of feature, request genaamd in de simulatie, bestaat meestal uit meerdere deeltaken waarbij een deeltaak onder een bepaalde categorie valt (type A, B en C). Ook worden er prioriteiten toegewezen aan de requests. Zo bestaan de requests met hoogste prioriteit vaak uit meerdere deeltaken ten opzichte van de requests met minder hoge prioriteit. Verder was het zo dat de vier teams opgedeeld waren op basis van specialisatie. Bij team rood was er een expert van onderdeel type A, bij team blauw was er een expert van onderdeel type B maar deze was ook intermediaire voor type A en kon expert worden door samen een project af te werken met team rood. Tot slot was er team groen en team paars die elk gespecialiseerd waren in type C.



Figuur 33: Het visualisatiebord van de Okaloa flowlab simulatie

Deze simulatie werd tweemaal gedaan zodat de problemen zichtbaar werden in de eerste simulatie en hierop ingespeeld kon worden in de tweede simulatie. Wat er al snel duidelijk werd na de eerste simulatie, was dat er veel geblokkeerd werk was. Dit betekent dat de persoon die met een bepaalde taak bezig was vast zat en dus niet verder kon. Aangezien een feature uit deeltaken bestaat en er voor elke deeltaak een andere specialisatie nodig was, is er sprake van een 'process bottleneck'. Dit wil zeggen dat het geblokkeerd werk bij het ene team ervoor zorgt dat de andere teams hun werk niet kunnen afmaken. Verder is er ook sprake van een 'knowledge bottleneck' aangezien er een team was (team blauw) dat zich kon specialiseren in een type A deeltaak, maar door het knelpunt de kans niet heeft gekregen. Dit wordt voorgesteld in figuur 34 waarbij deze fenomenen zich voordoen. Verder was het ook zo dat er veel meer requests van prioriteit drie afgewerkt waren ten opzichte van deze van prioriteit één of twee.



Figuur 34: De arbeidsverdeling (bron van de agile coach)

De eerste reactie van de werknemers en wat vaak ook in de werkelijkheid gebeurt, is dat de limiet van WIP (work in progress) in zo een geval verhoogd wordt, of dat er extra mankracht wordt

aangenomen. Het probleem van de limiet van WIP te verhogen zorgt voor uitstel van het knelpunt, want zodra het eerste knelpunt is opgelost, komt er binnen de kortste keren een tweede aan het licht. Extra mankracht zou een oplossing kunnen zijn, maar is niet ideaal want wanneer het knelpunt zou zijn opgelost kan het zijn dat je met te veel mankracht zit waardoor er weer meer 'idle time' komt. Bovenstaande oplossing verhoogt tevens de WIP en verlaagt de voorspelbaarheid waardoor de kans op geblokkeerd werk verhoogd wordt. Verder zou er ook gekozen kunnen worden voor het specialiseren van bepaalde werknemers in een bepaald type werk, maar het grote nadeel dat hieraan is verbonden, is dat dit veel tijd en geld kost. Het moet vanuit een ander perspectief benaderd worden. Namelijk het knelpunt identificeren en de oorzaak hiervan oplossen.

Bij de tweede simulatie waren er twee nieuwe toepassingen, namelijk: team blauw en team rood mochten enkel aan een project samen beginnen (apprentencing) en team paars en team groen mochten in pair werken (pairing of mobbing). Het resultaat was dat team blauw zich al snel had gespecialiseerd in het type A onderdeel, en door pairing was het zo dat de experts van team groen en team paars samen aan projecten werkten waardoor ze hun kennis en inzichten konden delen. Zo wordt het probleem dat wanneer een expert langer aan een soort project heeft gewerkt, waardoor deze altijd voorrang krijgt of dat er meer vertrouwen is in deze persoon, ook van de kaart geveegd. Een ander resultaat na het simuleren van tien fictieve dagen was dat er een veel betere verdeling van het aantal requests met verschillende prioriteiten was. Dit toont aan dat collaboratie cruciaal is in een agile bedrijf maar ook het afstemmen van demand (requests) op de vaardigheden en capaciteit van de huidige teams. Het was ook zo dat de werknemers vertelden dat het simulatie spel goed in elkaar stak omdat ze er veel elementen in terugzagen die ook van toepassing zijn bedrijf X. De agile coach vertelde dat zo een visualisatie (de simulatie) mensen helpt om bottlenecks te vinden en om een beter beeld te krijgen over het gehele proces. "Observation as a basis!"

Er moet naar het geheel gekeken worden. Ten eerste moet de bekwaamheid verbeterd worden om aan de vraag te kunnen voldoen (downstream). Ten tweede wordt er geprobeerd om de noden van de klant te begrijpen, te anticiperen, vorm te geven en natuurlijk vraag te creëren (upstream/vraag). Hier gaat er geprobeerd moeten worden om de vraag vorm te geven om beter gebruik te maken van de bekwaamheid van de werknemers. Tot slot wordt er gekeken naar de verschillende opties en gaat er geprobeerd worden deze twee delen in de rooilijn te brengen. Zo wordt er een end-to-end flow van een vermoedelijke tot tevreden behoefte gecreëerd.

In het hoger onderwijs kan er ook sprake zijn van knelpunten van het proces en knelpunten van kennis. Door samen te werken kan er van anderen geleerd worden zodat diegene die 'gespecialiseerd' is in een bepaald iets zijn kennis kan delen met diegene die er minder over weet. Verder is het zo dat wanneer er samengewerkt wordt, er minder kans is op knelpunten van het proces aangezien er samen nagedacht wordt hoe het probleem opgelost moet worden. Dit kan gelinkt worden aan principe 5 aangezien motivatie, de bereidwilligheid van leren hier zeer belangrijk is. Ook principe 6 is van toepassing aangezien het delen van informatie zodat de anderen bijleren cruciaal is om de knelpunten omtrent kennis op te lossen. Principe 8 is hier ook belangrijk, want wanneer deze twee knelpunten opgelost worden, is er meer kans dat er constante ontwikkeling is en de 'idle time' verminderd wordt. Tot slot is principe 11 hier ook toepasbaar, want wanneer een team zelf knelpunten van proces en kennis kunnen oplossen, ervan uitgegaan kan worden dat het team zelfsturend leert te zijn. We

kunnen dus concluderen dat wanneer er volgens de agile principes gewerkt wordt, deze knelpunten geminimaliseerd kunnen worden

4.5 Beperkingen van deze studie en verder onderzoek

Deze studie werd gedaan in een relatief klein bedrijf dat volgens de agile principes werkt. Er werd gebruik gemaakt van observatie, maar ook werden er vragen over de werking van het bedrijf gesteld aan enkele werknemers. Elk bedrijf werkt en is anders op verschillende vlakken. Om een betere veralgemening te kunnen maken, zouden er in verschillende agile bedrijven observaties gedaan moeten worden. Op deze manier zouden de verschillen tussen de bedrijven geïdentificeerd kunnen worden en kunnen deze verder vergeleken worden met de hoger onderwijscontext. Verder zou er gebruik gemaakt kunnen worden van interviews om op deze manier verschillende meningen van werknemers te kunnen bekomen zodat niet louter het werkingsaspect van het bedrijf in acht genomen wordt.

4.6 Conclusie

Door een dag te observeren in bedrijf X, dat in het kort werd beschreven in sectie 4.2, konden de werking en elementen van het bedrijf achterhaald worden. De werking en elementen werden uitgelegd in sectie 4.3 en werden steeds vergeleken met het hoger onderwijs waardoor er enkele bevindingen in hoofdstuk 3 versterkt konden worden. Vervolgens werd in sectie 4.4 de inbreng van de agile coach aangehaald en werd deze inbreng beschouwd als aanvulling op de bevindingen van de observatie. We kunnen concluderen dat het bedrijf zich niet 100% aan de agile principes houdt, maar het is wel toepasselijk om te zeggen dat het bedrijf een agile bedrijf is na het inspecteren van de verschillende elementen. Verder zullen de verschillen tussen de hoger onderwijscontext en de bedrijfscontext samenvattend beschreven worden met mogelijke aanpassingen. Op deze manier kan de tweede deelvraag beantwoord worden, namelijk: "Hoe verschilt de hoger onderwijscontext van de bedrijfscontext op vlak van projectwerk?":

- **De werkomgeving:** De indeling van de werkvloer in bedrijf X is zo geïdentificeerd zodat er optimaal gewerkt kan worden, terwijl er in het hoger onderwijs enkel gebruik wordt gemaakt van leslokalen. Lokalen of plaatsen beschikbaar maken waar teamleden samen aan hun project/groepswork kunnen werken, bevordert de teamsfeer en zorgt ook voor inzichten van iedereen die aanwezig is.
- **Visuele management borden:** Een agile bedrijf kan voor elk departement en het overkoepelende, fysieke visuele management borden opzetten, terwijl in het hoger onderwijs het moeilijk is om deze borden ergens te stationeren. De oplossing hier is het werken met online tools. Dit zodat er steeds geweten is wie met wat bezig is, maar ook zodat er metingen (bijvoorbeeld: met behulp van burndown charts) gemaakt kunnen worden om te zien hoe het project vordert en om te zien of er zwakke schakels (individuen) in het project zitten die het project vertragen.
- **Gebruik van agile methode(s):** In een bedrijf worden verschillende methodes gebruikt, langs elkaar en door elkaar. In het hoger onderwijs wordt er in de meeste gevallen één methode aangeleerd die gebruikt moet worden voor de voltooiing van een project. Het aanleren van één methode is een goede start om te leren werken op een agile manier.

- **Uitstelgedrag:** In een bedrijf kan er sprake zijn van uitstelgedrag, maar dit wordt beperkt aangezien er van de werknemers verwacht wordt dat ze goed werk leveren en ze worden betaald (in het geval er slecht gepresteerd wordt, is er kans op ontslag). In het hoger onderwijs is er meer sprake van uitstelgedrag & free-riders, maar dit wordt beperkt door het gebruik van een agile methode. Het probleem is dat er, buiten het geven van slechte punten, geen sancties kunnen gegeven worden aan de studenten die niet goed presteren. Aanbevelingen voor studenten is het sorteren (triage) van de projecten die er moeten gedaan worden: Kijk wat er moet gebeuren, kijk of er zich hier grote struikelblokken kunnen voordoen en bepaal dan in welke mate het werk kan worden uitgesteld/verdeeld.
- **Dagelijkse communicatie:** In een bedrijf kunnen de werknemers dagelijks met elkaar samenwerken en kan er gemakkelijk face-to-face communicatie gedaan worden. In een hoger onderwijs is het moeilijker om de studenten dagelijks samen te laten komen, dagelijks aan een project te werken en dagelijks face-to-face te laten spreken met elkaar. Op regelmatige basis samenkomen is, afhankelijk van de werklast, voldoende.
- **Dagelijkse collaboratie en interactie:** In in een bedrijf kunnen de werknemers dagelijks samenwerken en zorgt dit ervoor dat ze van elkaar leren. Zoals hierboven vermeld is het moeilijk om de studenten dagelijks te laten samenwerken. Aangezien de werklast van één opleidingsonderdeel minder is dan de werklast van een project in een bedrijf is het dagelijks samenwerken niet nodig in het hoger onderwijs. Veel studenten werken het liefst zelfstandig (conclusie uit de inleidende enquête) aan een project, maar door samenwerking zullen deze studenten tot nieuwe inzichten komen en zal er van elkaar geleerd worden (met het idee dat de student naar de lessen is geweest en weet waarover het gaat).
- **Motivatie:** In een bedrijf wordt ervan uit gegaan dat de werknemers intrinsiek gemotiveerd zijn. In het hoger onderwijs is de grootste motivator het slagen om op deze manier een diploma te behalen (instrumentele waarde). Het uitbreiden van kennis (intrinsieke waarde) is ook een motivator bij de studenten, maar niet bij iedereen. Er zouden door het onderwijsteam manieren doorgevoerd worden om de studenten meer intrinsiek gemotiveerd te maken, zodat de prestatie tijdens een project verbetert.
- **Behandelen van onzekerheid:** In een bedrijf is bij het maken van een project 'waarom' en voor 'wie' bekend, maar moet de 'wat' en 'hoe' nog ontdekt worden. In een hoger onderwijs is 'wat' en 'hoe' al vaak bekend, maar is 'waarom' en voor 'wie' vaak niet bekend. Dit kan opgelost worden door te werken met real-life scenario's, waarbij de de studenten de waarom en wie ook te weten komen. Hierdoor zal de intrinsieke motivatie voor sommige studenten toenemen.
- **Specialisatie versus Leren:** In een bedrijf is er vaak sprake van specialisatie en deze wordt door het samenwerken doorgegeven aan elkaar. In een hoger onderwijs is er nog geen specialisatie, tenzij er een student zich in zijn vrije tijd bezighoudt met de methode die er gehanteerd wordt. In een hoger onderwijs is er meer sprake van leren zodat er in het verdere verloop gespecialiseerd kan worden. Verder is leren een traag proces dat niet op één twee drie gebeurd is. Constant met de zaak bezig zijn zorgt ervoor dat de struikelblokken op tijd gedetecteerd kunnen worden en er hiernaar aangepast kan worden. In het hoger onderwijs zou dit kunnen worden opgelost met tussentijdse project demos of deadlines.

Hoofdstuk 5: Formuleren van richtlijnen voor het toepassen van agile projectmanagement in het hoger onderwijs

In dit laatste hoofdstuk van deze thesis zullen al de bevindingen uit de drie databronnen, namelijk de literatuurstudie, de enquêtes en de observatie in bedrijf X, samengevoegd worden aan de hand van richtlijnen. Met deze richtlijnen moeten het onderwijsteam (en de studenten) rekening houden wanneer er gebruik wordt gemaakt van agile projectmanagement in het hoger onderwijs. Deze richtlijnen zullen onderbouwd worden door de agile principes en zijn toepasbaar op elk type projectwerk. Verder zijn deze richtlijnen niet verplicht om te volgen, maar zorgen deze ervoor dat enerzijds studenten beter zullen presteren en anderzijds het onderwijsteam weet waar de problemen zich zouden kunnen voordoen. De volgende tabel geeft een antwoord op de derde deelvraag, namelijk: “Welke wijzigingen en aanbevelingen op vlak van projectwerk in het hoger onderwijs kunnen er worden doorgevoerd om agile projectmanagement zo goed mogelijk toepasbaar te maken?”

Principe 1: “Onze hoogste prioriteit is het tevreden stellen van de klant door het vroegtijdig en voortdurend opleveren van waardevolle software.”

Onderwijscontext: Er moet gewerkt worden met tussentijdse deadlines van afgewerkte deeltaken (features uit de product backlog) van een project, die gecontroleerd worden door het onderwijsteam. Op deze manier kan er vroegtijdig feedback gegeven worden waardoor het eindproduct waardevoller is dan wanneer er gewerkt wordt met één finale deadline.

Richtlijn:

- Werk met sprints van twee weken (afhankelijk van de werklast) waarbij er op het einde van deze twee weken een afgewerkte deeltaak ingediend moet worden.

Principe 2: “Verwelkom veranderende behoeftes, zelfs laat in het ontwikkelingsproces. Agile processen benutten verandering tot concurrentievoordeel van de klant.”

Onderwijscontext: Om de studenten klaar te stomen voor een steeds veranderende omgeving in het bedrijfsleven, kunnen eventueel veranderende elementen in de opdracht verwerkt worden waardoor de studenten hierop moeten inspelen.

Richtlijnen:

- Maak gebruik van veranderingen tijdens het proces (wanneer de einddeliverable gekend is).
- Geef opbouwende kritiek op de tussentijdse deadlines zodat de studenten gemotiveerd zijn om zich aan te passen en aanpassingen zullen doorvoeren op een juiste manier.

Principe 3: “Lever regelmatig werkende software op. Liefst iedere paar weken, hooguit iedere paar maanden.”

Onderwijscontext: “Lever na elke sprint een werkend (voldoen aan de vooropgestelde criteria) deelproduct op.”

- Zorg dat je een lijst van criteria waaraan het deelproduct moet voldoen hebt gemaakt die verwerkt zit in de studieleidraad of als een apart bestand op het onderwijsplatform staat. Zo zal een team hier meer rekening mee houden.
- Geef na elke sprint constructieve feedback (aan de hand van de criteria), zodat de studenten na de aanpassingen het deel als werkend ('done') kunnen beschouwen.

Principe 4: "Mensen uit de business en ontwikkelaars moeten dagelijks samenwerken gedurende het gehele project."

Onderwijscontext: Studenten moeten op regelmatige basis met elkaar samenwerken gedurende het gehele project en moeten regelmatig feedback krijgen van het onderwijsteam om aan de noden van het onderwijsteam te voldoen.

Richtlijnen:

- Plan verplichte vaste momenten in in het lessenrooster waar studenten samen moeten werken aan het project.
- Motiveer studenten om buiten de vaste contacturen ook samen te komen om samen te werken aan het project.
- Geef op tijd (bijvoorbeeld na elke sprint) feedback aan de studenten zodat dit de 'klant'- 'bedrijf' relatie bevordert. Op deze manier weet de studenten wat er van hen verwacht wordt. Zorg er dus voor dat je niet traditioneel te werk gaat door enkel aan het begin en aan het einde van het project benaderbaar te zijn.
- Studenten: Probeer buiten de vaste contactmomenten een gezamenlijk vrij moment te vinden (desnoods na de schooluren). Als dit niet lukt voor iedereen, probeer dan met de teamleden die beschikbaar zijn samen te komen. Laat dit werk achteraf door de persoon die er niet was nakijken zodat deze op de hoogte is van wat er is gedaan of besproken.

Principe 5: "Bouw projecten rond gemotiveerde individuen. Geef hen de omgeving en ondersteuning die ze nodig hebben en vertrouw erop dat ze de klus klaren."

Onderwijscontext: "Zorg ervoor dat de studenten gemotiveerd zijn om aan een project te werken. Geef hen de omgeving en ondersteuning die ze nodig hebben en vertrouw erop dat ze de klus klaren."

Richtlijnen:

- Stel de groepen samen aan de hand van kwaliteiten van individuen (door middel van vragenlijsten die de studenten moeten invullen). Maak op basis van deze vragenlijsten een profiel van iedere student zodat deze door de teamleden bekeken kunnen worden en zodat onderling van iedereen geweten is wat de plus- en werkpunten zijn.
- Wanneer het opleidingsonderdeel gevolgd wordt door meerdere opleidingen, zorg er dan voor dat van elke opleiding minstens in één persoon in het team zit, zodat de problemen vanuit verschillende standpunten bekeken kunnen worden.
- Werk met groepen van 4 tot 6 personen, afhankelijk van de werklust, met een voorkeur voor 5 personen zodat er in groep meerderheidsbeslissingen genomen kunnen worden.
- Zorg ervoor dat de inhoud van het opleidingsonderdeel deels is afgestemd op de waarde en de verwachtingen van de studenten en zorg ervoor dat je bereikbaar bent. Op deze manier kan je motivatie creëren (voor extra voorbeelden om waarde en verwachtingen bij studenten te creëren, kijk naar sectie 2.8.5).

- Voorzie de studenten van een project dat gebaseerd is op een real-life voorbeeld, waardoor er interesse in het topic gecreëerd wordt en de bijhorende materie beter verwerkt wordt. Met real-life voorbeelden zal de de student ook weten 'waarom' het project gedaan wordt en voor 'wie' het bedoeld is.
- Probeer ervoor te zorgen dat free-riders (meelifters) vroegtijdig in het proces geïdentificeerd worden zodat deze hierop gewezen kunnen worden en zich kunnen aanpassen. Dit kan het best gedaan worden door maandelijkse peerevaluatie.
- Studenten: Wijs ineffectieve groepsleden erop dat ze beter hun best moeten doen. In het geval er niets veranderd, meld deze ineffectieve teamleden dan aan het onderwijsteam zodat deze de nodige maatregelen kan treffen.
- Bied studenten een geschikte ruimte aan op de vaste contactmomenten en eventueel op andere momenten, waardoor de studenten regelmatig een projectruimte ter beschikking hebben om samen te werken aan het project. Zorg ervoor dat de tafels in de vorm van werkeilanden staan.

Principe 6: "De meest efficiënte en effectieve manier om informatie te delen in en met een ontwikkelteam is door met elkaar te praten."

Onderwijscontext: Dit principe kan volledig overgenomen worden aangezien het communicatieaspect hier wordt aangehaald. Communicatie bij een project is zeer belangrijk en volgens de agile methodologie is face-to-face communicatie de beste manier van communiceren.

Richtlijnen:

- Studenten: Probeer zoveel mogelijk face-to-face te communiceren tijdens de momenten waar heel de groep samen is omdat dit tot meer inzichten leidt. Maak verder goede afspraken omtrent de communicatie buiten deze gezamenlijke momenten.
- Studenten: Deel kennis met elkaar en geef hints aan elkaar, zodat iedereen zich kan 'specialiseren' in een bepaald onderwerp.
- Biedt de studenten tools aan die ze kunnen gebruiken voor de bevordering van communicatie alsook de werking van het project. Zo zal bijvoorbeeld een visueel management bord helpen dubbelwerk te vermijden, zien wie met wat bezig is, welke taken er nog moeten gebeuren en waar er zich problemen bevinden (geblokkeerd werk). Verder kunnen visuele management tools er ook voor zorgen dat er metingen gedaan kunnen worden om bijvoorbeeld de vordering van het project te schetsen.

Principe 7: "Werkende software is de belangrijkste maat voor voortgang."

Onderwijscontext: "Een goede oplossing of benadering van de deeltaak van het project is de belangrijkste maat voor voortgang"

- Studenten: Zorg dat de deeltaken aan de bepaalde criteria voldoen, zodat deze als 'done' beschouwd kan worden en hier verder op gebouwd kan worden.

Principe 8: "Agile processen bevorderen constante ontwikkeling. De opdrachtgevers, ontwikkelaars en gebruikers moeten een constant tempo eeuwig kunnen volhouden."

Onderwijscontext: "Studenten moeten aandachtig en gemotiveerd zijn vanaf dag 1 zodat er geen achterstand wordt opgelopen en meelifters vroegtijdig gedecteerd worden. Op basis van de werklast moeten de studenten een constant tempo kunnen volhouden"

Richtlijnen:

- Werk met tussentijdse deadlines om uitstelgedrag te voorkomen.
- Wijs studenten erop dat achterop geraken een grote invloed zal hebben op het verdere verloop van het project.
- Geef richtlijnen aan studenten over de hoeveelheid tijd ze per sprint aan het project moeten besteden, zodat ze weten hoeveel tijd ze er aan moeten benutten en deze tijd beschouwen als het tempo waaraan ze constant moeten kunnen werken.

Principe 9: "Voortdurende aandacht voor een hoge technische kwaliteit en voor een goed ontwerp versterken agility."

Onderwijscontext: "Heb voortdurend aandacht voor de kwaliteit van het project. Volg de gegeven richtlijnen omtrent kwaliteit zorgvuldig na."

Richtlijnen:

- Geef feedback omtrent de kwaliteit van het werk, waardoor studenten zich hiernaar kunnen aanpassen.
- Bepaal op voorhand criteria waaraan de tekst moet voldoen, zodat de studenten zich hierop kunnen baseren (in de studieleidraad of op het schoolplatform).
- Studenten: Heb voortdurend aandacht voor de kwaliteit van het product door het werk regelmatig te vergelijken met de criteria die vooraf zijn vastgelegd (door de student zelf of door het onderwijsteam).

Principe 10: "Eenvoud, de kunst van het maximaliseren van het werk dat niet gedaan wordt, is essentieel."

Onderwijscontext: "Hou je werk concreet en beknopt en zorg ervoor dat de werking van het team optimaal verloopt."

Richtlijnen:

- Leg een paginalimiet op waardoor het team verplicht is om het rapport van het werk beknopt te houden.
- Wijs de student erop dat de tekst of het werk zo concreet mogelijk gehouden wordt en dat hiermee in het finale rapport rekening wordt gehouden bij de puntenverdeling.
- Studenten: Maak gebruik van sprint retrospective waarin wordt nagedacht over de werking van het team en hoe deze verbeterd kan worden. Dit zal ervoor zorgen dat de problemen vanuit andere perspectieven bekeken worden en er geen onnodig (foutief) werk gedaan wordt. Op deze manier kan er uiteindelijk (in het beste geval) de beste oplossing gebruikt wordt.
- Studenten: Probeer de tekst zo concreet mogelijk te houden. Zorg dat de uitleg van een bepaald onderwerp beperkt is en herhaling vermeden wordt "Liever weinig maar goed, dan veel maar slecht."

Principe 11: "De beste architecturen, eisen en ontwerpen komen voort uit zelfsturende teams."

Onderwijscontext: Dit principe kan helemaal overgenomen worden. De leider die meestal de mensen leidt, moet veranderen naar een leider die het proces leidt (zoals een ScrumMaster).

Richtlijnen:

- Voorzie de studenten van voldoende uitleg wanneer een project gestart wordt. Dit kan aan de hand van een inleidende les over het project, uitleg over de manier van aanpak, uitleg over de elementen van de methode om interpretatieproblemen te voorkomen en eventuele extra bronnen die studenten zelfstandig kunnen doornemen.
- Ga na of de studenten al kennis hebben van agile methodes door met de docenten van voorgaande opleidingsonderdelen te communiceren. Zo ja, geef dan alleen een inleidende les zoals hierboven vermeld wordt. Zo nee, voorzie de studenten dan van voldoende achtergrond zodat ze de gehanteerde methode op een juiste manier toepassen.
- Maak een onderscheid tussen projecten in de bachelor en in de master. Deze in de bachelor mogen meer begeleid zijn en in de master wordt er verwacht dat studenten op een zelfstandigere basis te werk gaan bij een project.
- Studenten: Wanneer er zich proces knelpunten voordoen, probeer dit eerst binnen het team op te lossen of spreek andere teams aan alvorens er naar het onderwijsteam wordt gegaan.
- Studenten: Werk met een leidersfiguur die het proces leidt. Deze persoon moet zeggen wat er moet gebeuren maar moet niet zeggen wat de teamleden moeten doen. Deze functie kan afgewisseld worden zodat elk teamlid leert omgaan met het leiden van een proces.

Principe 12: "Op vaste tijden, onderzoekt het team hoe het effectiever kan worden en past vervolgens zijn gedrag daar op aan."

Onderwijscontext: Dit principe kan in het hoger onderwijs, wanneer er volgens een agile methode gewerkt wordt, ook volledig overgenomen worden.

Richtlijnen:

- Net zoals bij principe 10, benadruk dat de studenten een sprint retrospective (zoals bij Scrum) houden na het voltooien van een sprint. Dit zal ervoor zorgen dat de studenten nadenken over de werking van het project en zullen hierdoor sneller geneigd zijn aanpassingen door te voeren zodat de werking verbeterd wordt.
- Probeer ervoor te zorgen dat alle elementen van een agile methode in acht genomen worden.
- Studenten: Maak gebruik van de sprint retrospective nadat er feedback is gegeven van het onderwijsteam na bijvoorbeeld een sprint.
- Studenten: Maak gebruik van een 'daily' Scrum meeting aan het begin van het moment wanneer er samengekomen wordt, zodat de problemen van de voorgaande dagen aangepakt kunnen worden.

Tabel 4: De richtlijnen voor het toepassen van agile projectmanagement in het hoger onderwijs

Verschillende richtlijnen zijn vanzelfsprekend, maar moeten wel in acht genomen worden. Wanneer deze richtlijnen gehanteerd worden, is de kans op een succesvol project, waar er gewerkt wordt volgens een agile methode, groter. Er kan hier geconcludeerd worden dat wanneer een onderwijsteam gebruik wil maken van agile projectmanagement in een project, ze het project hiernaar moeten vormen met de bekomen richtlijnen die hierboven staan weergegeven. Verder zal

niet alleen het onderwijsteam aanpassingen moeten doorvoeren, maar ook studenten zullen zich naar deze richtlijnen moeten aanpassen. Dit betekent dat er zowel voor het onderwijsteam als de student meer tijd en moeite in het project zal moeten gestoken worden.

Hoofdstuk 6: Conclusie

In deze masterthesis werd er onderzoek gedaan naar de toepasbaarheid van agile projectmanagement (APM) in het hoger onderwijs. Als onderzoeksmethode werd er gebruik gemaakt van een exploratieve case study waarbij er drie databronnen zijn gebruikt om de richtlijnen die in hoofdstuk vijf staan weergegeven te formuleren. Ten eerste werd er in de literatuurstudie onderzocht wat projectmanagement is en werd er dieper ingegaan op agile projectmanagement. Vervolgens zijn er in de literatuurstudie studies geanalyseerd omtrent agile projectmanagement in het hoger onderwijs. Op basis van de bevindingen in deze studies, de algemene elementen van APM en risico's verbonden aan APM, zijn er vragen opgesteld die in de enquêtes, waarvan de resultaten beschreven staan in hoofdstuk drie, werden gebruikt. Met de resultaten uit de enquêtes konden de twaalf leidende agile principes vertaald worden naar het hoger onderwijs en konden deze verder onderbouwd worden met de bekomen resultaten. Hieruit werden richtlijnen gevormd om agile projectmanagement beter toepasbaar te maken in het hoger onderwijs. Deze richtlijnen zijn terug te vinden in hoofdstuk vijf en staan steeds bij het bijhorende oorspronkelijke en vertaalde principe weergegeven. Vervolgens is er in hoofdstuk vier een observatie uitgevoerd in een agile bedrijf, om op deze manier de verschillen op vlak van projectwerk met het hoger onderwijs te identificeren, maar ook om de richtlijnen die er gevormd zijn uit hoofdstuk drie te versterken en aan te vullen. Doordat de bevindingen uit de drie databronnen samengevoegd zijn tot richtlijnen in hoofdstuk vijf, kan er een antwoord op de centrale onderzoeksvraag geformuleerd worden, namelijk: "Op welke manier kan agile projectmanagement toegepast worden in het hoger onderwijs?". Agile projectmanagement kan toegepast worden in het hoger onderwijs als er rekening wordt gehouden met de richtlijnen die geformuleerd zijn in hoofdstuk vijf. Deze richtlijnen zijn voornamelijk bedoeld voor docenten, maar sommige ook voor studenten. De richtlijnen zijn niet verplicht om te volgen, maar zorgen er wel voor dat studenten zich beter zullen inzetten aangezien de meerderheid van deze richtlijnen gebaseerd zijn op hun voorkeuren. Verder zorgen deze richtlijnen er ook voor dat docenten zich beter kunnen voorbereiden en komen ze te weten wat er van hen verwacht wordt wanneer er volgens een agile methode gewerkt wordt. Ze moeten de structuur van hun opleidingsonderdeel zo aanpassen dat het mogelijk is om agile projectmanagement toepasbaar te maken. Er zal dus zowel van de studenten als van de docenten veel inspanning gevraagd worden om agile projectmanagement succesvol te implementeren in het hoger onderwijs. Tot slot kunnen deze richtlijnen in toekomstig onderzoek verder gevalideerd worden en kunnen deze richtlijnen eventueel gebruikt worden om een aangepaste agile methode te ontwikkelen die speciaal bedoeld is voor projectwerk in het hoger onderwijs.

Literatuurlijst:

- Ambrose, S.A., Bridges, M.W., DiPietro, M., Lovett, M.C. and Norman, M.K. (2010). *What factors motivate students to learn?: How Learning Works: Seven Research-Based Principles for Smart Teaching*. San Francisco: Jossey-Bass.
- Awad, M. A. (2005). A comparison between agile and traditional software development methodologies. Honours program thesis, *University of Western, Australia*.
- Balaji, S., & Murugaiyan, M. S. (2012). Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, 2(1), 26-30.
- Banica, L., Radulescu, M., Rosca, D., & Hagi, A. (2017). Is DevOps another Project Management Methodology? *Informatica Economica*, 21(3), 39-51.
- Beck, K. (1999). Embracing change with Extreme Programming. *IEEE Computer*, 32(10), 70-77.
- Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). *Manifesto for Agile Software Development*. Geraadpleegd op 15 december 2017, van <http://agilemanifesto.org/>
- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), 61-72.
- Borysowich, C. (2007). *Creating Regression Test Suites*. Geraadpleegd op 3 april 2018, van <https://it.toolbox.com/blogs/craigborysowich/creating-regression-test-suites-071507>.
- Cockburn, A. (2000). Selecting a project's methodology. *IEEE software*, 17(4), 64-71.
- Delhij, A., van Solingen, R., & Wijnands, W. (2015). *The eduScrum guide*. Geraadpleegd op 27 november 2017, van http://eduscrum.nl/en/file/CKFiles/The_educrum_Guide_EN_1,2.
- Dubakov, M., & Stevens, P. (2008). Agile tools: The good, the bad and the ugly. *Report, TargetProcess, Inc*.
- Fernandez, D. J., & Fernandez, J. D. (2008). Agile project management—agilism versus traditional approaches. *Journal of Computer Information Systems*, 49(2), 10-17.
- Fitsilis, P. (2008). Comparing PMBOK and Agile Project Management software development processes. In *Advances in Computer and Information Sciences and Engineering*, pp. 378-383, Springer, Dordrecht.
- Hass, K. B. (2007). The blending of traditional and agile project management. *PM World Today*, 9(5), 1-8.
- Ionel, N. (2008). Critical analysis of the scrum project management methodology. *Annals of the University of Oradea, Economic Science Series*, 17(4), 435-441.
- Jimenez, O. & Cliburn, D. (2016). Scrum in the undergraduate Computer Science Curriculum. *Journal of Computing Sciences in Colleges*, 31(4), 108-114.
- Karlesky, M., & Vander Voord, M. (2008). Agile Project Management. *ESC*, 247(267), 4.
- Kroll, P., & Kruchten, P. (2003). *The rational unified process made easy: a practitioner's guide to the RUP*. Addison-Wesley Professional.
- Landry, J., & McDaniel, R. (2016). Agile Preparation Within a Traditional Project Management Course. *Information Systems Education Journal*, 14(6), 27-33.
- Lang, G. (2017). Agile learning: sprinting through the semester. *Information Systems Education Journal*, 15(3), 14-21.
- Mahnic, V. (2010). Teaching Scrum through Team-Project Work: Students' Perceptions and Teacher's Observations. *International Journal of Engineering Education*, 26(1), 96-110.

- Marks, D. (2002). *Development Methodologies Compared*. Geraadpleegd op 20 februari 2018, van www.ncycles.com.
- Maurer, F. & Martel, S. (2002). Extreme Programming: Rapid Development for Web-Based Applications. *IEEE INTERNET COMPUTING*, 6(1), 86-90.
- May, J., York, J., & Lending, D. (2016). Play ball: Bringing Scrum into the Classroom. *Journal of Information Systems Education*, 27(2), 87-92.
- Melnik, G., & Maurer, F. (2005, May). A cross-program investigation of students' perceptions of agile methods. Proceedings of the International Conference on Software engineering, 481-488.
- Project Management Institute. (2017). *PMBOK® Guide – Sixth Edition*. Project Management Institute.
- Rajamanickam, D. (2005). Successful Project Management Using Agile Methodology. *Journal of the quality assurance institute*, 19(3), 15-18.
- Rico, D.F. (2008). What is the ROI of agile vs. traditional methods? An analysis of extreme programming, test driven development, pair programming, and scrum (using real options). *TickIT International*, 10(4), 9-18.
- Rico, D. F., & Sayani, H. H. (2009). Use of agile methods in software engineering education. Proceedings of the 2009 *Agile Conference*, 174-179.
- Schwaber, K. & Sutherland, J. (2016). *The Definitive Guide to Scrum: The Rules of the Game*. geraadpleegd op 10 januari 2018, van <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf#zoom=100>.
- Schwalbe, K. (2012). Managing a Project Using an Agile Approach and the PMBOK® Guide. *Proceedings of the Information Systems Educators Conference*, 2167, p. 1435.
- Silverman, D. (2005). *DOING QUALITATIVE RESEARCH*. London: SAGE publications Ltd.
- Stoica, M., Mircea, M., & Ghilig-Micu, B. (2013). Software Development: Agile vs. Traditional. *Informatica Economica*, 17(4), 64-76.
- Sutherland, J. (2010). *Agile principles and values*. Geraadpleegd op 20 februari 2018, van <http://msdn.microsoft.com/en-us/library/dd997578.aspx>.
- Takpuie, D. & Maureen, T. (2016). Investigating the Characteristics Needed by Scrum Team Members to Successfully Transfer Tacit Knowledge During Agile Software Projects. *The Electronical Journal Information Systems Evaluation*, 19(1), 36-54.
- Valle, M., & O'Mara, K. J. (2015). Adaptive Project Management: A Classroom Exercise to Explore the Fundamentals of Agile (Scrum). *Journal of the Academy of Business Education*, 16(1), 257-275.
- VersionOne Inc. (2016). *10th Annual State of Agile Survey*. Geraadpleegd op 26 februari 2018, van <http://www.agile247.pl/wp-content/uploads/2016/04/VersionOne-10th-Annual-State-of-Agile-Report.pdf>
- Walczak, W., & Kuchta, D. (2013). Risks characteristics of agile project management methodologies and responses to them. *Operations Research & decisions*, 23(4), 75-95.
- Wideman, R. M. (2002). Comparing PRINCE2® with PMBoK®. *AEW Services, Vancouver, BC, Canada*.

Bijlages

1) De resultaten van de vragenlijst aan het einde van elke sprint in de studie van Mahnic (2010)

		Sprint 1			Sprint 2			Paired sample t-test (t-value)
		Mean	Std. dev.	One-sample t-test (t-value)	Mean	Std. dev.	One-sample t-test (t-value)	
1	Clarity of requirements specified in the Product Backlog Was the Product Backlog clear enough? Did the short description of each requirement suffice to understand what the Product Owner really wanted?	3.21	0.98	1.14	3.87	0.90	5.28**	3.91**
2	Effort estimation Were the estimates of workload provided in the Product Backlog adequate?	2.97	0.98	-0.19	3.80	0.81	5.44**	4.25**
3	Maintenance of the Sprint Backlog Was it clear how to maintain the Sprint Backlog and provide data requested by the spreadsheet application?	3.72	0.75	5.19**	4.30	0.79	8.96**	4.31**
4	Administrative workload The administrative work requested by the Scrum method does not represent a significant additional workload.	3.24	1.15	1.13	3.27	1.08	1.35	0.49
5	Co-operation with the ScrumMaster Was the co-operation with the ScrumMaster adequate?	4.07	0.92	6.24**	4.33	0.71	10.27**	2.20*
6	Co-operation with the Product Owner Was the co-operation with the Product Owner adequate?	3.79	0.90	4.74**	4.03	0.85	6.66**	1.44
7	Co-operation with other Team members Was the co-operation with other Team members adequate? Does the Scrum method encourage co-operation?	4.00	1.04	5.20**	4.10	0.92	6.53**	0.46
8	Workload Was the amount of work required on the project adequate?	3.76	0.99	4.14**	3.73	0.78	5.12**	0.00
9	Satisfaction with the work on the project Are you satisfied with the work on the project?	3.72	0.80	4.89**	3.80	0.92	4.74**	0.96
10	Satisfaction with the Scrum Method Is the method useful? Would you recommend the method to other software developers?	3.72	1.07	3.66*	3.87	0.94	5.07**	0.50

* $p < 0.05$; ** $p < 0.01$

2) inleidende enquête agile voor het onderzoek in hoofdstuk 3

Inleidende enquête Agile projectmanagement

Deze enquête wordt in het vak Agile Business Model Design afgenomen door de studenten die dit vak volgen. Deze enquête zal gebruikt worden voor onderzoek voor de thesis van een laatste jaar student handelsingenieur in de beleidsinformatica omtrent agile projectmanagement in hoger onderwijs. De informatie vergaard uit deze enquête zal strikt gebruikt worden voor deze doeleinden. Er wordt verzekerd dat de respondent anoniem zal blijven en dat persoonlijke gegevens niet verder verspreid zullen worden. De bedoeling van deze inleidende enquête is om te kijken wat de student al weet over 'agile' methodes en om te weten te komen of er al agile praktijken worden toegepast in de

projectwerken die de student moet maken gedurende de studieloopbaan. Verder zullen er persoonlijke meningen gevraagd worden over bepaalde onderwerpen met betrekking tot groepswerken/projecten.

1. PERSOONLIJKE GEGEVENS

1.1. Wat is je studentnummer?

.....

1.2. Welke studierichting volg je?

BI Informatica

1.3. Wat is je leeftijd?

.....

1.4. Wat is je geslacht?

M V

2. AGILE

2.1. Heb je al ooit gehoord van agile software development? Zo ja, vanwaar ken je het?

JA NEE Uitleg:

2.2. Als het bovenstaande antwoord ja was (zo nee, ga naar deel 3), welke agile software development methodologie(ën) ken je?

- | | |
|--|--|
| <input type="checkbox"/> SCRUM | <input type="checkbox"/> FDD (Feature-Driven Development) |
| <input type="checkbox"/> XP (Extreme Programming) | <input type="checkbox"/> Crystal Clear |
| <input type="checkbox"/> DSDM (Dynamic Systems Development Method) | <input type="checkbox"/> RAD (Rapid Application Development) |
| <input type="checkbox"/> ASD (Adaptive Software Development) | <input type="checkbox"/> RUP (Rational Unify Process) |
| <input type="checkbox"/> Andere: | |

2.3. Heb je goede ervaringen met de toepassing van deze methodes? Wat zijn de voor- en/of nadelen?

JA NEE Voor-/Nadelen:

2.4. Zou je andere mensen aanraden om zich te verdiepen in één van bovenstaande methodes?

JA NEE Waarom?:

3. ALGEMEEN

3.1. Welke factoren hebben volgens jou een negatieve invloed op het maken en voltooien van een groepswerk of project?:

3.1.1. Onvoldoende uitleg van de prof over de manier van aanpak (met het idee dat je zelf opzoek werk doet naar het onderwerp):

sterk mee eens eens neutraal oneens sterk mee oneens

3.1.2. Meelifters (geen actieve meewerking van bepaalde groepsleden):

sterk mee eens eens neutraal oneens sterk mee oneens

3.1.3. Iemand die de leiding neemt en delegeert:

sterk mee eens eens neutraal oneens sterk mee oneens

3.1.4. Aantal groepsleden > 4 personen:

sterk mee eens eens neutraal oneens sterk mee oneens

3.2 Rangschik de antwoorden van de volgende vragen (1= meest toepasselijk)

3.2.1. Op welke manier kan de prof de studenten het best voorbereiden op een groepswerk of project (met het idee dat een master student zelfstandig kan werken)

- Uitgebreide uitleg van het groepswerk geven (met methodes van uitwerking)
---	-------

- Referenties naar bronnen met extra informatie over het desbetreffende onderwerp
- Inleidende les met uitgebreide informatie over het onderwerp
- De studenten zoeken zelf info naar het onderwerp en krijgen in de week na de opzet van het groepswerk een Q&A moment waarbij de prof antwoorden en aanvullende informatie geeft
- Andere:

3.2.2. Hoe kunnen we het probleem van meelifters aanpakken zodat iedereen actief mee werkt aan een groepswerk of project?

- Met tussentijdse deadlines werken waarop punten staan
- Wekelijkse peer evaluation
- Maandelijkse peer evaluation
- Het melden van ineffectieve groepsleden aan de prof
- De tekst of uitwerking per persoon in een verschillende kleur zetten
- Tussentijdse testen om te checken of de student het onderwerp goed begrijpt
- Andere:

3.2.3. Welke manier verkies je om te communiceren met de groepsleden?

- Via een gesprek of groep op facebook
- Via een tool (zoals slack)
- Afspreken met de groepsleden om alles te bespreken
- Telefonisch
- Skype
- andere:

3.2.4. Welke manier verkies je voor het samenstellen van een groep?

- De groep zelf kiezen (zodat je samen zit met mensen die je (goed) kent)
---	-------

- De groep op een willekeurige manier laten samenstellen door de prof
- De groep samenstellen op basis van kwaliteiten van individuen (bepaald door een individuele vragenlijst over het individu) zodat een groep bestaat uit mensen met verschillende kwaliteiten
- Andere:

3.3.1. Iemand die de leiding neemt in een groep en de groep op een goede manier stuurt is een veel voorkomend fenomeen. Denk je dat zonder leider een groepswork tot een goed einde kan gebracht worden?

JA NEE

Leg uit:

3.3.2. Wat is volgens jou de ideale grootte van een project groep? Leg uit.

3.3.3. Heb je liever tussentijdse deadlines of één finale deadline zodat je zelf kan plannen wanneer je aan het project werkt?

tussentijdse deadlines één finale deadline

Waarom?

3.3.4. Stel er is 1 deadline, heb je dan de neiging om uit te stellen, er zo snel mogelijk aan te beginnen of het werk te verspreiden?

- uitstellen asap beginnen het werk verspreiden

Waarom?

3.3.5. Verdeel je liever het werk zodat je er zelfstandig aan kan werken of werk je er liever als groep aan?

- individueel in groep

Waarom? Van welke factoren hangt dit af?

3.3.6. Vind je dat er een onderscheid moet zijn tussen groepswerken/projecten in de bachelor en groepswerken/projecten in de master?

- JA NEE

Op welk vlak?

Bedankt voor de medewerking!

3) De tweede enquête voor het onderzoek van hoofdstuk 3

VERVOLG ENQUÊTE AGILE PROJECTMANAGEMENT

Als vervolg op de inleidende enquête die werd afgenomen in week 1, is dit een tweede enquête die in week 6 zal worden afgenomen. Ook hier zal de informatie strikt gebruikt worden voor onderzoek omtrent agile projectmanagement in het hoger onderwijs voor de thesis van een laatste jaar student handelsingenieur in de beleidsinformatica. Er wordt verzekerd dat de respondent anoniem zal blijven en dat persoonlijke gegevens niet verder verspreid zullen worden. Deze enquête heeft als doel te weten te komen of er bepaalde ideeën of overtuigingen gewijzigd zijn sinds de start van het project.

1. PERSOONLIJKE GEGEVENS

1.1. Wat is je studentnummer?

.....

1.2. Welke studierichting volg je?

BI Informatica IW Fysica

1.3. In welk jaar zit je?

.....

1.4. Wat is je leeftijd?

.....

1.5. Wat is je geslacht?

M V

2. AGILE

2.1. Hoe ervaar je momenteel de werking van het project? Maakt jullie groep goede vooruitgang richting het voltooien van het project? Verklaar.

2.2. Van welk van de volgende elementen/handelingen maak je gebruik/zijn zichtbaar gedurende het project? (Meerdere opties zijn mogelijk)

- Sprints van 1-4 weken met deliverable die vastgelegd wordt door het onderwijsteam
- Tussentijdse feedback van het onderwijsteam
- Voldoende interactie/communicatie met het onderwijsteam (vragen stellen/voldoende uitleg krijgen)
- Vastgelegde contactmomenten waarin jullie face-to-face samen kunnen werken in een lokaal
- Iemand in de groep die ervoor zorgt dat alles goed verloopt
- Product backlog (een bestand waarin alle features/taken staan die er voltooid moeten worden)
- Bepalen aan het begin van de sprint welke features/taken er afgewerkt moeten worden gedurende de desbetreffende sprint
- Bij het samenkomen starten met een korte vergadering waarin besproken wordt wat er gedaan is, wat er moet gebeuren en wat er beter kan
- Aan het einde van een sprint een moment waar er gekeken gaat worden naar hoe het team gedurende de sprint heeft gepresteerd en hoe de werking in de volgende sprint beter kan lopen
- De hele groep bestaat uit gemotiveerde individuen
- De groep is cross-functioneel (er zijn meerdere specialiteiten)
- Werken met een tool (zoals trello) waarbij je een burndown chart (progressie grafiek) kan maken
- Je features/taken zijn altijd klaar op het einde van een sprint
- Het onderwijsteam heeft ervoor gezorgd dat jullie voldoende informatie hebben gekregen om aan het project te kunnen beginnen

2.3 Als je de meerderheid van de vakjes (≥ 10) bij de voorgaande vraag hebt aangevinkt, kan je zeggen dat je groep volgens de principes van scrum werkt (zo nee, ga naar 2.4). Heb je goede ervaringen met het toepassen van deze methode in het project? Denk je dat deze manier van werken voor een betere productiviteit zorgt dan wanneer je één finale deadline hebt en werkt op je eigen manier? Licht toe en ga dan naar 2.5.

JA NEE Licht toe:

2.4 Aan het begin van het project is er verteld dat jullie volgens de principes van scrum zouden moeten werken. Wat is de reden dat je dit/deze element(en)/handeling(en) bij vraag 2.2 niet hebt aangeduid? Denk je dat sommige elementen/handelingen niet haalbaar zijn in een onderwijscontext? Ben je van mening dat sommige van deze elementen/handelingen niet nodig zijn? Heb je geen weet van sommige elementen/handelingen?

2.5. Denk je dat deze manier van werken toepasbaar is op alle type projecten? zo nee, welke type projecten niet?

JA NEE

2.6. Wat is volgens jou de ideale lengte van een sprint in een project voor een opleidingsonderdeel?

3. ALGEMEEN

3.1. Op welke manier wordt er het meest gewerkt binnen jullie project? Licht toe.

Individueel In groep Evenwicht tussen beide

Licht toe:

3.2. Hoe communiceert jullie groep? (meerdere opties zijn mogelijk).

- | | |
|---|--|
| <input type="checkbox"/> Via een facebook groep/gesprek | <input type="checkbox"/> Telefonisch |
| <input type="checkbox"/> Via een tool (zoals slack) | <input type="checkbox"/> Wanneer jullie samenkomen |
| <input type="checkbox"/> Per mail | <input type="checkbox"/> Skype |
| <input type="checkbox"/> Andere: | |

3.3 Is dit de meest efficiënte manier volgens jou? Leg uit.

3.4. Welke van de volgende problemen zijn al aan bod gekomen in jullie groep? Leg uit hoe je dit probleem/deze problemen hebt aangepakt.

- | | |
|---|--|
| <input type="checkbox"/> Meelifers (ongemotiveerde teamleden) | <input type="checkbox"/> Geen beslissingen in groep kunnen nemen |
| <input type="checkbox"/> Iemand neemt alles op zich | <input type="checkbox"/> Je kan het niet goed vinden met iemand |
| <input type="checkbox"/> Geen gezamenlijk vrij moment vinden | <input type="checkbox"/> Iemand neemt de opdracht niet serieus |
| <input type="checkbox"/> Andere: | |

Leg uit:

3.5. De groep is samengesteld aan de hand van de kwaliteiten van individuen. Is dit een betere manier dan het zelf samenstellen van de groep?

JA NEE Waarom?

3.6. Geef aan in welke mate je het met de volgende stellingen eens/oneens bent:

3.6.1 Door samen te werken in dezelfde ruimte bekom je meer inzichten over een bepaald probleem

sterk mee eens eens neutraal oneens sterk mee oneens

3.6.2. Samenwerken in dezelfde ruimte zorgt voor vertraging van het project (door bv: afleiding)

sterk mee eens eens neutraal oneens sterk mee oneens

3.6.3. Een leider die zegt wat de teamleden moeten doen heeft een positieve invloed op een project

sterk mee eens eens neutraal oneens sterk mee oneens

3.6.4. Een student is gemotiveerd om goed te presteren omdat hij/zij een goede score wil behalen

sterk mee eens eens neutraal oneens sterk mee oneens

3.7. Soms zit je vast en weet je even niet wat je moet doen om verder te geraken. Hoe pak je dit probleem aan? (rangschik: 1 = meest toepasselijk, 5 = minst toepasselijk)

- Je stelt je vraag aan medestudenten
- Je stelt je vraag aan de prof (per mail of persoonlijk)
- Je houd je in afwachting met andere taken bezig
- Je doet niets en wacht tot de volgende les
- Andere:

3.8. Welke factoren motiveren jou/kunnen jou motiveren om goed te presteren voor een project? (rangschik: 1 = meest toepasselijk, 8 = minst toepasselijk) Als er andere zaken zijn die jou motiveren of zouden kunnen motiveren, twijfel dan niet om ze neer te schrijven!

- Slagen
- Hoge punten behalen
- Je kennis uitbreiden zodat je meer ontwikkeld bent voor de bedrijfswereld
- Je weet de reden waarom het project wordt gedaan en voor wie het dient (Je maakt deel uit van een groter geheel. Bv: onderzoek proffen & onderzoek bedrijven)
- Het project is een uitdaging
- Een leuke groep met een leuke sfeer
- Je maakt je ouders hiermee gelukkig
- Andere:

Bedankt voor je medewerking!

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Agile projectmanagement in hoger onderwijs: onderzoek en toepassing

Richting: **master in de toegepaste economische wetenschappen:
handelsingenieur in de beleidsinformatica**
Jaar: **2018**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Taravella, Anthony

Datum: **1/06/2018**