

Mislukte IT projecten: een kwestie van beter plannen?

T. Stamper

June 22, 2009

Abstract

In deze scriptie wordt bestudeerd of het mogelijk is om, met behulp van de planning, de kwaliteit en het nut van IT projecten positief te beïnvloeden. Dit wordt gedaan door te bekijken hoe een project is opgebouwd. Met behulp van een bekende theorie wordt duidelijk gemaakt hoe de onderdelen van een project zich tot elkaar verhouden. Daarna wordt uitgewerkt wat er onder kwaliteit en nut van een project verstaan wordt. Vervolgens worden verbeteringen besproken voor projecten, die in de literatuur zijn genoemd. Tot slot worden deze verbeteringen in verband gebracht met de planning en suggesties gegeven voor verbetering.

Contents

1	Inleiding	3
1.1	Probleemstelling	3
1.2	Verantwoording	4
1.3	Theoretisch kader	5
1.3.1	Kennisgebied	5
1.3.2	Inperkingen	5
1.3.3	Concepten en theorie	6
1.4	Methode	7
1.4.1	Onderzoeksfunctie	7
1.4.2	Onderzoeksstructuur	7
1.5	Behandeling van de deelvragen in de scriptie	8
2	Het concept project	9
2.1	Het software ontwikkelingsproces	9
2.2	Agile ontwikkelingsmethoden	11
2.3	De theorie van P.S. Seligmann	11
2.4	De theorie van Seligmann en het software ontwikkelingsproces	13
2.5	Generalisatie	16
3	De kwaliteit en het nut van een project	17
3.1	De kwaliteit van een project	17
3.2	Het nut van een project	20
3.3	De kwaliteit en het nut in het geïntegreerde model	21
3.4	Generalisatie	21
4	Verbeteringen van de kwaliteit en het nut van een project	23
4.1	Verbeteringen van de kwaliteit	23
4.2	De kwaliteitsverbeteringen in het model	25
4.3	Verbeteringen van het nut	25
4.4	De verbeteringen van het nut in het model	26
4.5	Generalisatie	26

5	De verbeteringen toegepast op de planning	27
5.1	De planning	27
5.2	De planning in het model	28
5.3	De verbeteringen	28
5.4	De verbeteringen en de planning	28
5.5	Een alternatief	30
5.6	Verbeteringen van de planning voor een willekeurig type IT project	31
6	Conclusie	32
7	Planning	38
7.1	Globale planning	38
7.2	Herziene planning	39
7.3	Afspraken met de scriptiebegeleider	39
8	Logboek	40

Chapter 1

Inleiding

1.1 Probleemstelling

Regelmatig is op het nieuws te horen of in de krant te lezen dat er onderzoek gedaan is naar IT projecten. Vaak gaat het over het slagen danwel mislukken van projecten of de kosten danwel de duur van de projecten. In veel van deze berichten wordt gesproken over de uitloop van een project, oplopende kosten of hoge percentages van projecten die mislukken. Daarnaast wordt het nut van een project weleens in twijfel getrokken. Indien projecten gelukt zijn, is soms de kwaliteit van de resultaten een punt van discussie: is het wat gewenst was of valt dit tegen? Enkele voorbeelden van veelbesproken projecten zijn de OV-chipkaart, de Betuwelijn en het Elektronische PatiëntenDossier (EPD).

Als reden voor het mislukken van projecten worden verschillende dingen genoemd. Naast redenen als een niet zorgvuldig uitgevoerde aanbestedingsprocedure of een slechte kosteninschatting wordt een slechte planning ook vaak als reden opgegeven. Dit zette mij aan het denken. Immers, de planning is een vast onderdeel van een project en heeft invloed op het verloop en daarmee ook de uitkomsten van een project. Een goede planning zou daarom kunnen leiden tot resultaten waar meer mee mogelijk is. Maar hoe kan bepaald worden wat 'goed' is en wat niet? En hoe kan dat vertaald worden in een planning? Is het eigenlijk wel mogelijk om de resultaten van IT projecten te verbeteren aan de hand van de planning? Deze vragen leidde mij tot het stellen van de onderzoeksvraag:

Kan een planning bijdragen aan een hogere kwaliteit en een beter nut van een IT-project en zo ja, hoe?

Als het mogelijk is om met behulp van de planning dit te bewerkstelligen dan zal het antwoord gevormd worden door een beschrijving van hoe de

planning eruit zal zien: hoe de planning moet worden opgebouwd, wat de punten zullen zijn die erin komen te staan en wat elk punt bijdraagt aan het verhogen van de kwaliteit van de uitkomsten van het project.

Als het antwoord op de onderzoeksvraag negatief is, zal worden uitgelegd om welke reden(en) het niet mogelijk is om met behulp van de planning een hogere kwaliteit van de uitkomsten en het nut van een IT project te bewerkstelligen. De knelpunten worden dan genoemd en er wordt beschreven in welke richting dan gekeken kan worden voor verbeteringen op het gebied van de kwaliteit en het nut.

1.2 Verantwoording

De planning is een vast onderdeel van een IT project. In de planning worden de activiteiten genoemd die voor het project moeten worden uitgevoerd. De resultaten van uitvoering van de activiteiten in de planning vormen de uitkomsten van het project. Het nut van het project is mede afhankelijk van hoe (goed) de activiteiten zijn uitgevoerd.

Één van de belangrijke punten voor de bepaling van de waarde van de uitkomsten, is de kwaliteit. In de IT is hier iets voor, SQA (Software Quality Assurance). SQA omvat alle activiteiten die de kwaliteit van een project kunnen bevorderen. Echter, de IT is niet de enige: voor elk type project bestaat wel iets om de kwaliteit te controleren of te verhogen.

Voorbeeld: de bouw van een huis moet zodanig zijn dat voldaan wordt aan eisen voor brandveiligheid. Daarnaast zijn er standaarden op het gebied van de bouw. Deze standaarden hebben betrekking op gebruikte bouwmaterialen, de gebruikte bouwmethode etc. De standaarden zorgen ervoor dat het resultaat (het huis) een bepaalde 'basiskwaliteit' heeft. De geldende eisen en standaarden bevorderen dus de kwaliteit van een huis.

Er is veel te lezen over wat er onder kwaliteit verstaan wordt in SQA. Er worden o.a. punten in genoemd die voor elk project van toepassing zijn met betrekking tot de kwaliteit. Het is alleen niet (altijd) duidelijk wanneer en waar er op deze punten gecontroleerd kan worden. Dit terwijl de kwaliteit van een project zich ontwikkelt naarmate een project vordert. Dit duidt erop dat het wenselijk is dat er regelmatig gecontroleerd wordt op de punten willen de uitkomsten een zekere kwaliteit bevatten.

De planning van een project is een onderdeel waarmee deze regelmatige controle kan plaatsvinden. De onderzoeksvraag is dus relevant. Immers, zoals hierboven al genoemd is, is kwaliteit een belangrijk aspect voor IT projecten en men is gebaat bij elke mogelijkheid waarmee de kwaliteit van projecten positief kan worden beïnvloed.

1.3 Theoretisch kader

Voor deze scriptie zijn een aantal concepten van belang. Deze concepten worden beschreven onder het kopje 'Concepten en theorie'. Daarnaast wordt toegelicht op welk vlak de onderzoeksvraag te maken heeft met het kennisgebied (IT). Ook worden enkele inperkingen van de onderzoeksvraag in de scriptie gemaakt. Eerst zal het kennisgebied waarbinnen de onderzoeksvraag valt, aangegeven worden. Daarna volgt een stukje over inperkingen die zijn gemaakt en tot slot worden een drietal belangrijke concepten uitgelegd.

1.3.1 Kennisgebied

Binnen de IT bestaat een vakgebied dat gaat over de ontwikkeling van software. Dit vakgebied of anders gezegd, het ontwikkelen van software, is beter bekend onder de naam Software Engineering. Software Engineering omvat het hele proces van het opstellen van requirements tot de implementatie (en het onderhoud) van software. Typisch gebeurt dit ontwikkelen in de vorm van een project dat bestaat uit verschillende activiteiten. Een vast onderdeel van een project is de planning. Een van de punten die onderdeel uitmaken van het ontwikkelproces, is de kwaliteit en de controle hierop. De onderzoeksvraag past binnen deze punten.

1.3.2 Inperkingen

Het is moeilijk om voor alle soorten projecten die men kan onderscheiden, de onderzoeksvraag uit te werken. In principe zou de onderzoeksvraag ook gesteld kunnen worden door mensen in de bouw, gezondheidszorg of ergens anders. Er is daarom besloten om te kijken naar projecten in de IT, aangezien de IT het toekomstig kennisgebied is van de auteur van de scriptie. Binnen de verschillende type IT projecten die er zijn, is gekozen voor software ontwikkeling. Dit is gedaan omdat het ontwikkelen van software het meest bekende type IT project is en de uitwerking van de onderzoeksvraag het meest geschikt is voor dit type IT projecten. Andere typen zijn bijvoorbeeld de migratie van een systeem (dat gebruikmaakt van software van type X)

naar een ander systeem (dat gebruikmaakt van software van type Y), het construeren van een nieuw systeem dat bestaat uit een aantal andere systemen door bijv. het samenvoegen van databases/gegevens (zoals bij het EPD) of bijv. een toepassing van een nieuwe architectuur (bijv. SOA).

Het is mogelijk om op verschillende manieren de kwaliteit en het nut van projecten te verhogen, maar niet alle verbeteringen zijn IT-gerelateerd. Er is gekozen voor het bekijken van IT-gerelateerde verbeteringen. Hiermee wordt bedoeld, verbeteringen die betrekking hebben op de IT en door de IT kunnen worden aangebracht in een project. Binnen alle IT-verbeteringen die te noemen zijn, is gekozen voor het behandelen van verbeteringen op het gebied van de planning. Enerzijds is dit gedaan in verband met de omvang van de scriptie en de beschikbare tijd, anderzijds in verband met het feit dat de planning vaak als reden wordt genoemd voor het mislukken van projecten, zoals al was gebleken uit de probleemstelling.

Als er meer tijd was geweest, was het mogelijk om te onderzoeken of de manier van aanpak (bij software ontwikkeling zou dat kunnen zijn het gebruik van een agile methode of de waterval methode) invloed heeft gehad op de kwaliteit van de resultaten. Ook zou het mogelijk zijn geweest om te onderzoeken of de beslissingen die gemaakt zijn tijdens de uitvoer van de verschillende activiteiten correct zijn geweest. Een voordeel hiervan zou zijn dat de bevindingen eventueel het antwoord op de onderzoeksvraag kracht kunnen bijzetten, bijv. als het niet door deze punten kan komen dat bepaalde IT projecten slecht bruikbare, kwalitatief lage resultaten hebben.

1.3.3 Concepten en theorie

- **software project:** een software project wordt in deze scriptie gezien als een instantie van een software ontwikkelingsproces, dat bestaat uit gerelateerde activiteiten die tezamen resulteren in (bruikbare) software.
- **software ontwikkelings proces:** een set van gerelateerde activiteiten die tezamen resulteren in (bruikbare) software. De term 'software engineering proces' zal ook worden gebruikt in de scriptie; deze term heeft dezelfde betekenis en zal worden afgekort tot SE-proces.
- **Seligmann theorie:** theorie die de structuur van een modelleringsmethode beschrijft. In deze theorie worden een aantal elementen van een

modelleringsmethode beschreven: een way of thinking, way of modelling, way of working, way of controlling en een way of supporting. Als toevoeging wordt veelal nog een way of communicating onderscheiden.

1.4 Methode

1.4.1 Onderzoeksfunctie

De scriptie heeft twee onderzoeksfuncties: beschrijven en ontwerpen.

1.4.2 Onderzoeksstructuur

Om in staat te zijn de onderzoeksvraag te beantwoorden, is de onderzoeksvraag opgedeeld in een aantal kleinere (deel)vragen:

1. *Wat is een project?*

Deze deelvraag is nodig om vast te stellen waaruit een project bestaat. Het gaat niet zozeer om de definitie van een project, maar eerder om het concept project. De beantwoording van deze deelvraag zal een conceptueel model bevatten van alle onderdelen waaruit een project bestaat en hun relaties. Voor het uitwerken van deze deelvraag zal gebruikgemaakt worden van een theorie in de IT, die P.S Seligmann heeft ontwikkeld.

2. *Wat is de kwaliteit (en het nut) van een project?*

Om te kunnen bepalen of de kwaliteit en het nut van een project beïnvloed kunnen worden, is het belangrijk om te weten wat er wordt verstaan onder deze termen. Hierbij gaat het in principe om definities, maar er zal een link worden gelegd met de theorie van Seligmann.

3. *Kunnen de kwaliteit en het nut van een project positief worden beïnvloed?*

Het antwoord op deze deelvraag is belangrijk, omdat met deze deelvraag (in combinatie met het antwoord op de voorgaande deelvraag), vastgesteld kan worden op welke punten van een project, de uitkomsten positief kunnen worden beïnvloed. Het antwoord op de laatste deelvraag is afhankelijk van het antwoord op deze deelvraag.

4. *Hoe kan ervoor gezorgd worden dat projecten positief worden beïnvloed met behulp van de planning?*

Indien de voorgaande deelvraag positief beantwoord wordt (ja, dit kan), is het natuurlijk interessant om te weten hoe de planning kan bijdragen hieraan. Het is immers het onderdeel van een project wat behandeld wordt in de onderzoeksvraag.

De antwoorden op de (deel)vragen vormen samen het antwoord op de onderzoeksvraag.

1.5 Behandeling van de deelvragen in de scriptie

De deelvragen zullen in deze hoofdstukken worden besproken:

Deelvraag	Hoofdstuk
deelvraag 1	Hoofdstuk 2: Het concept project
deelvraag 2	Hoofdstuk 3: De kwaliteit en het nut van een project
deelvraag 3	Hoofdstuk 4: Verbeteringen van de kwaliteit en het nut van een project
deelvraag 4	Hoofdstuk 5: De verbeteringen toegepast op de planning
onderzoeksvraag	Hoofdstuk 6: Conclusie

Chapter 2

Het concept project

Om de onderzoeksvraag te kunnen beantwoorden is het allereerst van belang om een begrip te vormen van het concept project. Het type IT project wat behandeld zal worden is het software ontwikkelingsproject. Hiervan worden de algemene onderdelen benoemd en in een model weergegeven. Daarna wordt de Seligmann theorie uitgelegd en de link gelegd tussen de theorie en het software (ontwikkelings) proces. Ter verduidelijking van de theorie zal een veelgebruikt model worden getoond. In een ander model zal vervolgens worden weergegeven waar en hoe de theorie van Seligmann te plaatsen is in het software proces. Tot slot wordt de uitwerking van dit onderdeel vertaald naar een uitwerking voor het algemene geval (een willekeurig type IT project).

2.1 Het software ontwikkelingsproces

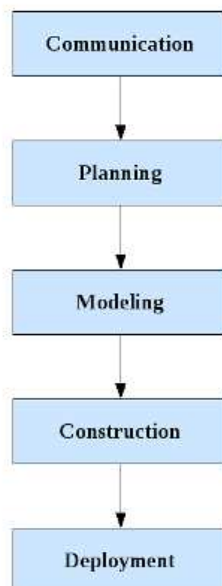
Binnen software engineering zijn verschillende ontwikkelingsmethoden bekend. Elke methode heeft zijn voor- en nadelen. Alle ontwikkelingsmethoden bevatten een bepaalde basisindeling voor de uitvoer van een software project. Ze zijn gegenereerd uit een zogeheten *process framework*[17]. Hierin zijn de basisactiviteiten die van toepassing zijn op alle ontwikkelingsmethoden, beschreven. De volgende basisactiviteiten zijn te onderscheiden:

- **Communication:** bij deze activiteit horen o.a. het klantoverleg, het achterhalen van de eisen voor de te ontwikkelen software (requirements engineering) en overige gerelateerde activiteiten m.b.t. de communicatie binnen een software project

- **Planning:** deze activiteit gaat over de inrichting van een software project na de communication fase. De taken die moeten worden uitgevoerd worden gedefinieerd, work products / deliverables worden gedefinieerd, de bronnen die nodig zijn voor de uitvoer van het project worden genoemd en de planning van de op te leveren delen (deliverables) wordt gemaakt
- **Modelling:** in deze activiteit wordt een model gemaakt (of meerdere modellen) om de vereisten (requirements) in beeld te brengen en beter te begrijpen en er wordt een ontwerp gemaakt voor de software
- **Construction:** deze activiteit bestaat uit het realiseren van de software: het genereren van de code valt hieronder. Daarnaast bestaat deze activiteit uit het testen van de software
- **Deployment:** onder deze activiteit valt de oplevering van de software (of een deel daarvan) aan de klant en de feedback die de klant geeft op de opgeleverde software. Ook ondersteuning voor de software (de opzet en/of uitrol daarvan) behoort tot deze activiteit

Schematisch ziet dit er als volgt uit:

Generic process framework for software projects



Afleidend uit het bovenstaande kan worden geconcludeerd dat een software project een instantie is van het generic process framework. De ontwikkelingsmethode van een software project bepaalt de invulling van de beschreven onderdelen en het traject dat hierdoor ontstaat, wordt doorlopen.

2.2 Agile ontwikkelingsmethoden

Ontwikkelingsmethoden als het waterval model gaan ervanuit dat requirements en functionaliteit die software moet bevatten van tevoren zijn vast te leggen en niet veranderen, echter de praktijk wijst uit dat dit niet altijd het geval is, zoals vermeld in [17]. Agile ontwikkelingsmethoden erkennen dat requirements steeds aan verandering onderhevig zijn en dit vertaalt zich ook in de invulling van de activiteiten uit het framework. In plaats van een invulling van elk onderdeel voor het hele project (alle requirements), focust de invulling zich op een specifiek onderdeel (een klein aantal requirements). Agile development is daarmee in feite een 'lichte' vorm van software engineering, aangezien het framework toegepast wordt op een subset van requirements: het maken en afleveren van kleine stukjes van de software [17]. Communicatie is het belangrijkste bij agile ontwikkelingsmethoden; er wordt immers weinig documentatie gegenereerd. *Planning* bij agile methoden bestaat (bijv. bij Xtreme Programming) uit het genereren van user stories, waarin specifieke requirements (bijv. bepaalde functionaliteit) van de te ontwikkelen software beschreven worden. Agile modelleringstools worden gebruikt om dit te visualiseren; dit vormt de invulling van de '*Modelling*' activiteit uit het framework. *Construction* bestaat uit het genereren en testen van de code van het stukje software en *Deployment* bestaat uit de oplevering aan de klant en de feedback die de klant geeft op het opgeleverde onderdeel. Het generic process framework is dus ook van toepassing op agile software ontwikkelingsmethoden.

2.3 De theorie van P.S. Seligmann

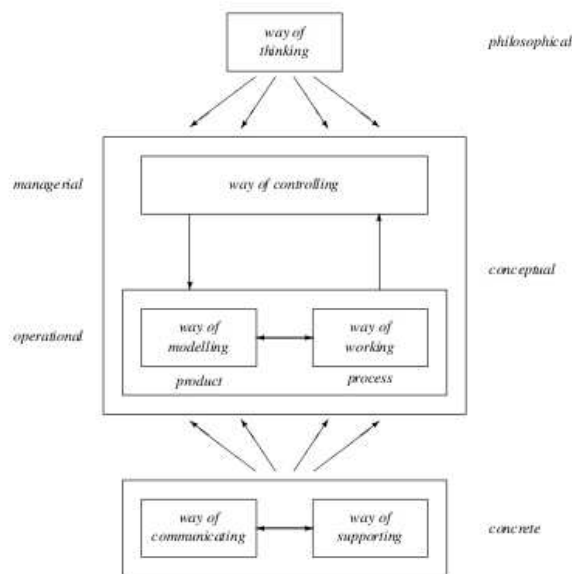
De theorie van Seligmann is een manier om gestructureerd naar informatiesysteem modelleringsmethoden te kijken. De theorie omvat een framework waarin 5 belangrijke onderdelen van een modelleermethode worden benoemd [19]. Deze onderdelen worden ook wel *ways* genoemd. De volgende onderdelen zijn in dit framework opgenomen: *way of thinking*, *way of working*, *way of modelling*, *way of supporting*, *way of controlling*. Vaak wordt ook een *way of communicating* genoemd als aanvulling op het framework[7][18]. De

onderdelen zijn hieronder beschreven:

- **way of thinking:** vrij vertaald is dit de kijk op de wereld. In de way of thinking worden de aannames, het (probleem) domein, oplossingen e.d. beschreven. Dit onderdeel vormt de filosofie achter een modelleringsmethode
- **way of working:** dit onderdeel wordt gebruikt om de manier waarop de beschrijving van een informatiesysteem in termen van de way of modeling tot stand komt, te beschrijven. (Deel)taken worden onderscheiden, een ordening wordt aangebracht en de wijze waarop taken worden uitgevoerd, wordt vermeld
- **way of modelling:** een abstracte beschrijving van de modelconcepten, met een beschrijving van de relaties en eigenschappen die hierbij horen. Dit onderdeel bevat een abstracte taal waarin een modelleringsmethode wordt uitgedrukt
- **way of controlling:** de managementkant van een modelleringsmethode. Deze beschrijft de way of working vanuit bestuurlijk perspectief. O.a. HRM, evaluaties en het projectmanagement vallen hieronder
- **way of supporting:** ondersteuning van de modelleringsmethode. Tools (bijv. CASE-tools) e.d. vallen binnen dit onderdeel
- **way of communicating:** dit onderdeel beschrijft hoe de abstracte concepten, zoals beschreven in de way of modelling, worden gevisualiseerd c.q. gecommuniceerd naar mensen. Meestal is dit in de vorm van een grafische notatie, bijv. ORM. ORM (Object-Role Modeling) beschrijft de communicatie binnen het applicatiedomein als een grammatica. Deze grammatica wordt uitgebreid met universele taalregels (ORC, Object Role Calculus)

De *way of communicating* en de *way of modelling* samen worden ook wel de *modelleertechniek* genoemd.

In het volgende, veelgebruikte, model is weergegeven hoe de onderdelen zich tot elkaar verhouden:



2.4 De theorie van Seligmann en het software ontwikkelingsproces

Het in de vorige twee secties geïntroduceerde framework en model, lijken op het eerste gezicht op zichzelf te staan. Het generic process framework (in deze sectie vanaf nu 'framework' genoemd) wordt gebruikt voor een beschrijving van software ontwikkelingsmethoden en het model van Seligmann als een structurele benadering van modelleermethoden. Toch is er een connectie te maken. Het model van Seligmann kan toegepast worden op het framework of concreet: de onderdelen van het framework kunnen geplaatst worden in het model van Seligmann. Dit helpt om de relaties tussen de verschillende onderdelen uit het framework beter te begrijpen. Hieronder is dit uitgewerkt:

Communication

In het framework wordt een onderdeel *Communication* genoemd, waaron-

der alle communicatie binnen een software project wordt verstaan. In het model van Seligmann wordt een *way of communicating* onderscheiden, waarmee bedoeld wordt: de manier waarop de modelleerconcepten uit de way of modelling worden gecommuniceerd naar anderen. Vaak is dit in de vorm van een grafische notatie. Deze notatie maakt duidelijk aan anderen hoe een domein, waarvoor een systeem wordt ontwikkeld, in elkaar steekt. Op eenzelfde wijze wordt in een software project d.m.v. praten, discussiëren en overleggen met elkaar, duidelijk gemaakt wat er gedaan moet worden voor het project. Het onderdeel communication past daarom het beste bij de *way of communicating* uit het model van Seligmann.

Planning

In het framework gaat dit onderdeel over de inrichting van het project: o.a. de taken die moeten worden uitgevoerd en de planning voor de op te leveren onderdelen vallen hieronder. Kijkend naar het Seligmann model zijn er twee onderdelen te noemen waarbij het onderdeel planning hoort: enerzijds de *way of working* (immers, hierin worden de taken onderscheiden, een ordening aangebracht voor het te maken informatiesysteem etc.) en anderzijds de *way of controlling* (het projectmanagement zorgt ervoor dat een planning wordt gemaakt, dat taken worden opgedeeld, m.a.w. het projectmanagement richt een project in). Het onderdeel planning past dus het beste bij deze twee onderdelen van het Seligmann model.

Modelling

In het framework wordt hiermee de de activiteit van het visualiseren van de requirements voor de software aangeduid en het ontwerp van de software. In het Seligmann model is een *way of modelling* opgenomen. Dit onderdeel staat voor de concepten die worden onderscheiden en een abstracte taal incl. relaties en eigenschappen die onderkend worden. Dit wordt gedaan om de relaties tussen concepten in een (probleem)domein beter te begrijpen. Op analoge wijze gebeurt dat bij het onderdeel *Modelling* uit het framework, maar dan met de requirements voor de te ontwikkelen software. Dit onderdeel past daarom het beste binnen de *way of modelling*.

Construction

Onder dit onderdeel wordt in het framework verstaan: het genereren van de code c.q. de software en het testen hiervan. Kijkend naar het model

van Seligmann, is er op het eerste gezicht geen onderdeel uit het model te noemen wat direct te koppelen is aan dit onderdeel. Echter, als er wat beter nagedacht wordt, biedt de *way of working* alsnog een mogelijkheid tot connectie. Immers, in de *way of working* is o.a. uitgewerkt hoe een systeem totstandkomt. Code wordt niet zomaar ineens gegenereerd, maar stapsgewijs, telkens komt er een stukje bij. Gedeeltelijk is de connectie met de *way of working* ook te verklaren aan de hand van de connectie van het onderdeel *Planning* met de *way of working*. Het realiseren van de software gaat aan de hand van een planning, die in het Seligmann model in de *way of working* is opgenomen.

Deployment

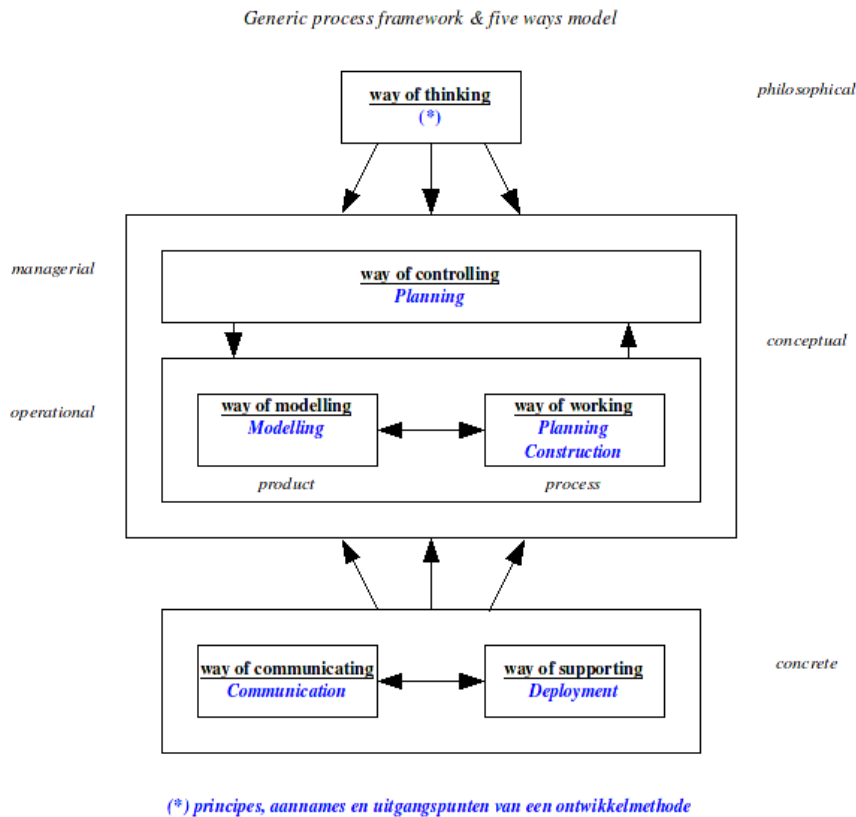
In het framework viel het opleveren van de software, de feedback van de klant en de opzet en uitvoer van de ondersteuning voor de software onder dit onderdeel. In het model van Seligmann is een *way of supporting* opgenomen. In het geval van het model wordt hieronder verstaan: de ondersteuning van de modelleermethode (en als voorbeeld werd genoemd: CASE tools). Als dit vertaald wordt naar software projecten, betekent dit de ondersteuning van de (opgeleverde) software. Het onderdeel deployment past daarom bij de *way of supporting*.

Way of thinking

De oplettende lezer merkt op dat inmiddels voor bijna alle onderdelen van het Seligmann model iets is ingevuld van het framework voor software projecten: het enige wat nog mist is een invulling voor de *way of thinking*. De logische vraag die nu volgt is: hoe zit het hiermee?

Uit de bespreking van de theorie van Seligmann in de vorige sectie, kwam naar voren dat onder de *way of thinking* de filosofie achter een modelleermethode verstaan werd. Aannames, het domein, oplossingen e.d. werden hierin opgenomen. Hiervoor bestaat er ook iets bij software ontwikkelingsmethoden. Deze methoden hebben elk een eigen '*kijk*' op hoe software totstandkomt. Dit vertaalt zich ook in de invulling van de genoemde onderdelen uit het framework. Onder de *way of thinking* wordt in de context van een software ontwikkelingsmethode verstaan: hoe kijkt een ontwikkelingsmethode aan tegen het SE-proces? Principes, aannames en opvattingen over (het verloop van) het proces e.d. vallen dus hieronder. Als voorbeeld kan gedacht worden aan het agile manifesto voor de agile development ontwikkelingsmethode: het manifesto is een concrete invulling van de *way of thinking* van die methode.

ode. Nu de connectie gemaakt is, kan dit in een model worden weergegeven. In een model ziet dit er zo uit:



2.5 Generalisatie

Gegeven de plaatsing van een software project in het Seligmann model, is het niet moeilijk om voor andere type IT projecten hetzelfde te doen. Een project is een instantie van het geïntegreerde model. De activiteitenindeling van een willekeurig IT project is te vergelijken met de indeling van activiteiten voor een software project: een IT project bevat een activiteit waarin uitgezocht wordt wat precies de bedoeling is van het project (vergelijkbaar met de activiteit 'Communication' bij software projecten), een activiteit waarin de te volgen aanpak wordt beschreven (zoals het onderdeel 'Planning'), een activiteit die (delen van) de aanpak visualiseert (zoals het onderdeel 'Modelling'), een activiteit die deze aanpak uitvoert (zoals 'Construction') en tot slot een activiteit waarin het resultaat wordt geïmplementeerd en het project wordt afgesloten (zoals bij 'Deployment').

Chapter 3

De kwaliteit en het nut van een project

Nu duidelijk is wat een project inhoudt, is het mogelijk om de onderwerpen kwaliteit en nut van een project te behandelen. Allereerst is het van belang om een definitie te geven van beiden. Vanuit de definitie kan dan gekeken worden hoe deze onderwerpen te plaatsen zijn in de context van een software project. Daarna zullen de onderwerpen in verband worden gebracht met het in het vorige hoofdstuk geïntegreerde model, bestaande uit de theorie van seligmann en het generic process framework. Tot slot worden de bevindingen op het gebied van een software project met betrekking tot de onderwerpen kwaliteit en het nut vertaald naar het algemene geval (een willekeurig type IT project).

3.1 De kwaliteit van een project

Wanneer er in een software project gesproken wordt over de kwaliteit van een project, dan wordt de kwaliteit van de software bedoeld die in het software project wordt ontwikkeld. Het gaat om de software kwaliteit. Er zijn verschillende definities van software kwaliteit, maar één daarvan is goed te gebruiken voor de beantwoording van de onderzoeksvraag, omdat hierin twee onderdelen genoemd worden die in dit hoofdstuk worden uitgewerkt:

”conformance to explicitly stated functional and performance requirements, explicitly documented development standards and implicit characteristics that are expected of all professional developed software[17].”

Om te kunnen bepalen of iets kwaliteit heeft kan, volgens de definitie, gekeken worden naar een drietal punten. Twee hiervan zijn van toepassing op de onderzoeksvraag:

1. nagaan in welke mate er is voldaan aan de requirements voor de software
2. nagaan in welke mate er is voldaan aan de impliciete, vaak niet genoemde, eisen aan de software (bijv. gebruiksgemak e.d.)

Alleen een methode om na te gaan hoe het gesteld is met deze punten, ontbreekt. Gelukkig bestaat er een methode om de punten te meten. Deze methode bestaat uit measures, measurement en indicatoren[17].

- **measure:** een kwantitatieve indicatie van de mate waarin een onderdeel van een systeem een bepaalde (gewenste) eigenschap bezit, bijv. een requirement van de software
- **measurement:** het proces van het verzamelen van waarden voor de verschillende measures
- **indicator:** 1 of meerdere measures die inzicht geven in het software project. Aan de hand van deze indicatoren kan een project manager of software engineer aanpassingen doen

De methode wordt toegepast op verschillende delen van een project: analyse, design, programmacode en testen van de software. Elk deel heeft zijn eigen specifieke indicatoren.

Voor elk deel van het ontwikkelingsproces waarvoor de methode wordt gebruikt, is het belangrijk om invullingen van de indicatoren te definiëren en een norm vast te stellen. Op z'n minst moet elke indicator 1 lage/slechte, 1 neutrale en 1 hoge/goede invulling hebben. Gekoppeld hieraan zou er een norm moeten worden vastgesteld die gehanteerd wordt voor het project. Wanneer de waardes voor de indicatoren onder de norm zitten, geeft dit reden om in te grijpen.

Voorbeeld: Om te bepalen of een persoon het naar zijn zin heeft op een feest, wordt gekeken naar de inhoud van het glas drinken van de persoon op het feest. De indicator voor 'het naar zijn zin hebben' is dus het glas drinken. Als invulling van de indicator worden de volgende waardes onderscheiden: leeg, voor een

kwart vol, halfvol, driekwart vol en vol. De norm die gehanteerd wordt is dat het glas minstens halfvol moet zijn, wil de persoon het naar zijn zin hebben. Door regelmatig te kijken naar de inhoud van het glas wordt het plezier van de persoon gemeten (= measurement). Zolang het glas driekwart vol of helemaal vol is, is alles goed. Maar is het glas halfvol, dan geeft dit aanleiding om in te grijpen (in dit geval, vraagt de persoon of het glas weer gevuld mag worden). De normwaarde dient dus als trigger voor het ondernemen van actie. Het overgaan van de indicator op een lagere waarde die nog boven de norm zit, is nog geen reden om in te grijpen. Het maakt echter wel *zichtbaar* dat de persoon het ietsje minder naar zijn zin heeft m.a.w. dat de waarde van de indicator aan het zakken is.

Indien er geen gebruik gemaakt wordt van de voorgestelde methode om te meten of voldaan is aan requirements van de software (zowel expliciete als impliciete) is het resultaat een slecht ontwikkeld systeem, met hoge onderhoudskosten[6][23]. Aangezien de business bepaalt of een project wordt uitgevoerd of niet en financiële zaken van belang zijn daarbij, is het daarom verstandig om de kwaliteit van het systeem ook uit te drukken in geld. Immers, zoals zal blijken uit het stuk over het nut van een project, is de (verwachte) toegevoegde waarde (business value) van een project de belangrijkste reden voor de business om een project al dan niet uit te voeren[20].

Aangezien er voor de verschillende onderdelen van een project indicatoren worden gebruikt, is het belangrijk om het overzicht te behouden tijdens de uitvoer van een project. Om inzicht te krijgen in de kwaliteit, zijn daarom regelmatige 'checks' nodig, momenten waarop een meting gedaan wordt van de verschillende indicatoren om in te zien of de kwaliteit gewaarborgd blijft, verhoogd wordt danwel verminderd. Hierbij komt de norm die van tevoren vastgesteld is, goed van pas.

In software engineering is een speciale rol weggelegd voor het management van een project om te zorgen voor een goede kwaliteit van software. In het projectmanagement is hiervoor iets in de vorm van Software Quality Assurance [17]. Software Quality Assurance (SQA) omvat de activiteiten die betrekking hebben op het plannen van SQA-activiteiten, het overzicht, documenteren, analyseren en rapporteren van de activiteiten. Vaak voert een aparte, onafhankelijke groep mensen dit werk uit: een SQA-groep. Het management probeert met SQA het niveau waarop de activiteiten in een project worden uitgevoerd, te bevorderen.

3.2 Het nut van een project

Het nut van projecten is soms een onderwerp van discussie[6][20], wanneer gekeken wordt naar de resultaten van een project. Vanuit een IT perspectief kan het resultaat van een project bevredigend zijn, bijv. omdat het ontwikkelingsproces goed verlopen is en als referentie kan worden gebruikt voor andere, vergelijkbare projecten (best practice) of omdat de software aan alle (kwaliteits)eisen voldoet. Dit hoeft niet te betekenen dat het project ook vanuit het business perspectief (het bedrijfsleven) een nuttig resultaat heeft: vaak worden er financiële indicatoren gebruikt om het effect van een project te meten. Dit heeft als gevolg dat, wanneer voor een project deze indicatoren slechte waardes bevatten, het project als geheel als 'niet nuttig/weinig bruikbaar' wordt bestempeld.

Wat uit het bovenstaande kan worden afgeleid, is dat het nut van een project twee kanten kent: de IT kant en de business kant[20]. Hoewel de IT kant zeker niet onbelangrijk is bij projecten, is het vaak de business kant waar uiteindelijk de meeste nadruk op wordt gelegd. Vaak wordt een project alleen goedgekeurd/uitgevoerd, als vermoedt wordt dat het een toevoeging zal zijn aan het bedrijf (bedrijfswaarde of bijv. vermindering van kosten).

Een probleem bij het bekijken van resultaten van projecten vanuit de business kant is het feit dat er, zoals hierboven al vermeld is, vrijwel uitsluitend financiële indicatoren worden gebruikt, zoals de return on investment (ROI)[14][20]. Soms zijn resultaten niet-financieel van aard bijv. een vergroting van de efficiency van activiteiten in een bedrijf. Bij het beoordelen van het nut van projecten moet een onderscheid gemaakt worden tussen financiële en niet-financiële vormen van toegevoegde waarde[20].

Ook op het gebied van het nut van een project kan de in dit hoofdstuk gepresenteerde meetmethode gebruikt worden. Op vergelijkbare wijze als bij de kwaliteit van een project is het dan mogelijk om na te gaan of het project nog wel het gewenste nut heeft (toegevoegde waarde). Indien dit niet (meer) het geval is, geeft dit reden om in te grijpen.

Er zijn verschillende momenten waarop het nut van projecten gemeten kan worden; voor het begin (in het projectvoorstel), in de ontwikkelingsfase, in de 'deployment' fase en tijdens het werken met het resultaat van het project (routine operation). Het is van belang om op meerdere momenten, het nut van een project na te gaan [20]. Op deze manier kan worden gezorgd dat het resultaat na de uitvoer van een project, ook nut heeft.

Ook hier is voor het management een belangrijke rol weggelegd. Het management kan het gebruik van indicatoren, het hanteren van normen en het overgaan tot actie wanneer normen niet worden gehaald, bevorderen door het toepassen van enkele principes van 'lean management'[1]. Een principe van lean management dat toe te passen is, is de variatie die in de kwaliteit van de deliverables zit, te verminderen door het ontwikkelen en invoeren van gestandaardiseerde procedures voor kwaliteitscontrole en regelmatige metingen van het nut. Daarnaast moet het management een projectteam overtuigen van het feit dat elke stap die genomen wordt in een project, kwaliteit en nut toevoegt aan het resultaat. Tot slot kan het management systemen invoeren die projectteams belonen als zij proberen het proces te optimaliseren en niet delen daarvan.

3.3 De kwaliteit en het nut in het geïntegreerde model

Wanneer de onderwerpen kwaliteit en nut in verband worden gebracht met het model uit het vorige hoofdstuk, zijn deze als volgt te plaatsen:

De kwaliteit van een project valt op te splitsen in de kwaliteit van de software en de acties die (door het management) worden uitgevoerd om te kunnen zorgdragen dat projecten een hoge kwaliteit bevatten. Dit onderwerp past binnen het geïntegreerde model onder het kopje *Planning/way of working* en *Planning/way of controlling* (vanwege de activiteiten die het management uitvoert).

Het nut van een project valt in te delen in het nut voor de IT en het nut voor de business (het bedrijfsleven). Omdat het gewenst is dat op verschillende momenten gekeken wordt wat het nut is, is het handig om deze in de planning te verwerken. Dit onderwerp past daarom het beste bij de *Planning/way of working* en *Planning/way of controlling*.

3.4 Generalisatie

De kwaliteit van een project kan opgesplitst worden in de kwaliteit van het resultaat van een project (bijv. kwaliteit van software) en de kwaliteit van de uitvoer van een project. Het nut van een project bestaat uit een businesskant en een IT kant. De businesskant is veelal bepalend voor het al dan niet

uitvoeren van een project. Om de kwaliteit en het nut te meten kan een methode die worden gebruikt die gebaseerd is op het gebruik van indicatoren. Deze indicatoren geven aan hoe het gesteld is met het project op een bepaald gebied van de kwaliteit of het nut. Er kan een onderscheid gemaakt worden in niet-financiële en financiële indicatoren. Om de methode goed te kunnen gebruiken, is het belangrijk om invullingen te definiëren (bijv. een vertaling naar onderhoudskosten of kosten van het uitvallen van een systeem) van de indicatoren en een norm vast te stellen. Door regelmatig de waarden van de indicatoren te meten, kan er tijdig worden ingegrepen, als blijkt dat een norm niet gehaald wordt.

Voor het waarborgen en/of bevorderen van de kwaliteit en het nut van een project, is een belangrijke rol voor het management weggelegd: het management kan hierbij gebruikmaken van principes van lean management. Standaardprocedures voor controle op kwaliteit en het nut kunnen ingevoerd worden, het projectteam moet overtuigd worden van het feit dat elke stap in het project iets toevoegt aan de kwaliteit en het nut en het management moet aansturen op optimalisatie van het proces.

Wanneer het bovenstaande wordt geplaatst in het geïntegreerde model, dan valt de kwaliteit en het nut van het resultaat van een project onder de *Planning/way of working* en vallen de activiteiten van het management op het gebied van bevorderen van de kwaliteit en het nut van een project, onder de *Planning/way of controlling*.

Chapter 4

Verbeteringen van de kwaliteit en het nut van een project

In het vorige hoofdstuk is ingegaan op de kwaliteit en het nut van een software project. In dit hoofdstuk zal stil worden gestaan bij verbeteringen, die in de literatuur genoemd worden op het gebied van deze twee punten. Bij zowel de verbeteringen op het gebied van de kwaliteit als het nut van projecten zal aangegeven worden waar de verbeteringen te plaatsen zijn in het geïntegreerde model uit het eerste hoofdstuk. Tot slot zal toegelicht worden of en hoe de gevonden verbeteringen gebruikt kunnen worden als verbeteringen voor een willeurig type IT project.

4.1 Verbeteringen van de kwaliteit

In het vorige hoofdstuk is kwaliteit besproken. Kwaliteit bestaat uit de kwaliteit van de software en de activiteiten die het management uitvoert om de kwaliteit van een project te beïnvloeden. Als nu gekeken wordt naar verbeteringen op het gebied van kwaliteit in de literatuur, valt op dat op het gebied van het projectmanagement vooral verbeteringen worden genoemd. Een tweetal verbeteringen, die typisch zijn voor het leeuwendeel van de voorgestelde verbeteringen[9][4][16][2], zijn hier uitgelicht.

[4] wijst erop dat veel voorgestelde verbeteringen van het software proces, zich focussen op een specifiek onderdeel. Het nadeel hiervan is dat het best kan zijn dat een verbetering op één gebied een achteruitgang op een ander gebied tot gevolg kan hebben. De ontwikkeling van software is een complex geheel en het is niet realistisch om te denken dat het op willekeurige plekken doorvoeren van verbeteringen een gunstig effect kan hebben op het

hele proces:

'Because software development is a complex system of interacting activities, random introduction of individual innovations cannot be expected to result in quantifiable improvements over the whole life-cycle.'

In het vorige hoofdstuk kwam dit ook al naar voren. In plaats daarvan zou op een hoger, conceptueel niveau gedacht moeten worden over verbeteringen. Volgens [4] moet het gehele proces beter gemanaged worden. Het verbeteren van het proces kan door de volgende 3 stappen te doorlopen:

1. het vaststellen van punten die vatbaar zijn voor verbetering
2. innoveren en evalueren om bepaalde verbeteringen voor de punten te ontdekken
3. de bevindingen overbrengen op de mensen (developers) die betrokken zijn bij een project bijv. d.m.v. trainingen

In [2] is een langdurige case study uitgevoerd in een software bedrijf. Hierbij wordt gebruikgemaakt van CMM, het Capability Maturity Model, wat aangeeft op welk niveau een bedrijf zit met betrekking tot het uitvoeren van software ontwikkelingsactiviteiten[17]. In het artikel wordt voorgesteld om een 'Quality Management System' in te voeren. De functie van dit systeem is het helpen verhogen van de kwaliteit van het werk door het management binnen een software bedrijf:

'Looking at the reference framework suggested by QMIM (=QMS), the organisation is able to understand the place and the interconnection of all elements that are important to deal with quality in a complete, unified and balanced way. Using this framework, the company can choose its own direction of development, depending on situational aspects such as maturity, specific business goals at a certain moment in time and business policy[2].'

De conclusie van [2] bevestigt de nadruk die in de literatuur gelegd wordt op verbeteringen van het management.

4.2 De kwaliteitsverbeteringen in het model

Uit de verbeteringen die genoemd worden in de literatuur en de twee uitgelichte voorbeelden, blijkt dat het merendeel zich focust op het management om de kwaliteit van de projecten te verbeteren. Deze verbeteringen zijn goed te plaatsen binnen het geïntegreerde model. Ze passen binnen het onderdeel *Planning/way of controlling*. Het management heeft immers de controle over een project en binnen het onderdeel *Planning* is een grote rol voor het management weggelegd.

4.3 Verbeteringen van het nut

Terugkomend op het vorige hoofdstuk, kan gezegd worden dat nut kan worden opgedeeld in het nut voor de IT en het nut voor het bedrijfsleven (business value). In het vorige hoofdstuk werd ook gezegd dat de nadruk lag op het laatste. Bij het zoeken naar verbeteringen op het gebied van het nut van projecten kan het beste gekeken worden naar verbeteringen die een gunstig effect hebben op de business value van projecten. In de literatuur is er weinig beschikbaar over verbeteringen op dit gebied: dit komt vanwege het feit dat business value in meerdere vormen aanwezig kan zijn in een project, zoals al bleek uit het vorige hoofdstuk. In de literatuur is wel wat te vinden over verbeteringen van business value met betrekking tot financiële measures, zoals de, in het vorige hoofdstuk al genoemde, ROI[14].

Een punt voor verbetering, dat afgeleid kan worden uit het vorige hoofdstuk, is het beter begrijpen c.q. uitwerken van de vormen van business value die een project (kan) bevat(ten). Daarbij is het belangrijk dat ze zodanig beschreven worden, dat ze meetbaar zijn, zoals benadrukt is in het vorige hoofdstuk.

Samenhangend hiermee, kunnen de business value meetwaarden meegenomen worden in het ontwerp van software.[14] gaat hier specifiek op in. Door het integreren van de modellering van de business value van een project in het ontwerp voor de software, kan beter gevisualiseerd worden, wat voor invloed keuzes hebben op de (uiteindelijke) business value van een project:

'Two major aspects of stakeholder value are adressed here. One is the business value to the development organisation stemming from software sales. Another is the value to the end-user stakeholder from varying sets and quality'....'Software process modeling and simulation can be used to reason about software value decisions.

It can help find the right balance of activities that contribute to stakeholder value with other constraints as cost, schedule or quality goals.'

In het artikel wordt gebruikgemaakt van een model waarin typische business value meetwaarden als kosten en ROI zijn opgenomen. Afhankelijk van bepaalde keuzes op het gebied van functionaliteit, levertijd, kwaliteit en andere gebieden, kan de business value worden berekend van software. Dit geeft inzicht in de gevolgen van keuzes voor de business value van een project.

4.4 De verbeteringen van het nut in het model

Gegeven het feit dat er wordt gesproken over modellering bij de verbeteringen die hier besproken zijn, kunnen de verbeteringen goed geplaatst worden in het geïntegreerde model. In het model passen deze verbeteringen bij het onderdeel *Modelling/way of modelling*.

4.5 Generalisatie

De verbeteringen van het nut (business value) en de kwaliteit zijn niet alleen van toepassing op software projecten, maar ook op IT projecten in het algemeen. Om een willeurig type IT project positief te beïnvloeden op het gebied van kwaliteit en nut, is het mogelijk om 1. het projectmanagement te verbeteren en voor de verbetering van het nut, 2. de business value bewust mee te modelleren in een project. Het management heeft de controle over een project en kan de kwaliteit van het proces en daarmee de resultaten positief beïnvloeden door bijv. lean management toe te passen. Het bewust meemodelleren van de business value in een IT project maakt zichtbaar hoe bepaalde keuzes in een project, invloed hebben op de (uiteindelijke) business value. Op deze manier kan de business value van een IT project positief beïnvloed worden.

Chapter 5

De verbeteringen toegepast op de planning

Tot nu toe zijn de volgende punten behandeld: het concept 'project', de kwaliteit en het nut van een project en verbeteringen op het gebied van de laatste twee punten.

Nu is het mogelijk om, gegeven de opgedane kennis uit de bespreking van de hierboven genoemde onderwerpen, specifiek te gaan kijken naar de planning. Allereerst zal duidelijk gemaakt worden waar de planning uit bestaat, wat zijn functie is en waar de planning te plaatsen is binnen het model. Daarna zullen de verbeteringen, genoemd in het vorige hoofdstuk, herhaald worden en in verband gebracht worden met de planning. De connecties worden geïnterpreteerd en er wordt bekeken of de genoemde verbeteringen verwerkt kunnen worden in de planning. Ook wordt een alternatief bekeken. Tot slot wordt een algemene verbetering gegeven die van toepassing is op elk type IT project.

5.1 De planning

De planning bevat de taken die moeten worden uitgevoerd om een project tot een goed einde te brengen. Deze taken zijn afgeleid en uitgewerkt in een eerder stadium van een project, meestal door het projectmanagement. In de planning worden de uit te voeren taken benoemd, de tijd en moeite die het gaat kosten worden vermeld (geschat) en er worden deadlines toegekend aan de taken. De functie van de planning is het werk dat gedaan moet worden, visualiseren en verdelen over de beschikbare tijd. Het biedt het management de mogelijkheid tot inzicht in de progressie van een project. Deze progressie

wordt uitgedrukt met behulp van indicatoren, op vergelijkbare wijze als dit bij de kwaliteit en het nut van een project wordt gedaan, maar dan met indicatoren specifiek voor de progressie (voor verdere details zie[17]).

5.2 De planning in het model

Gegeven de omschrijving en functie van de planning is het niet moeilijk om deze te plaatsen in het geïntegreerde model. De planning is een resultaat van de activiteit *Planning* en past daarom binnen de *Planning/way of working* (er staat in beschreven wat er gedaan moet worden, wat ervoor nodig is en wanneer het af moet zijn) en de *Planning/way of controlling* (het projectmanagement stelt de planning op en kan zien hoe een project vordert).

5.3 De verbeteringen

In het vorige hoofdstuk zijn een aantal verbeteringen genoemd voor de kwaliteit en het nut van een software project. Bij de verbeteringen voor de kwaliteit werd duidelijk dat het management van projecten beter kan. Als voorbeelden voor verbetering werden genoemd het volgen van een bepaalde methode door het management, de invoering van een "QMS" (Quality Management System) en/of het toepassen van 'lean management'. Op het gebied van het nut (business value) werden twee verbeteringen genoemd: het concreet beschrijven van de business value die een project bevat en het integreren van de modellering van business value in het ontwerp van de software.

5.4 De verbeteringen en de planning

Het management heeft een belangrijke rol in software projecten. Binnen het geïntegreerde model is het management (met name) verantwoordelijk voor de activiteiten bij de *way of working* en de *way of controlling*. De verbeteringen op het gebied van kwaliteit slaan voornamelijk op het functioneren van het management. De connectie die te maken is met de planning van een project zit hem in het feit dat de planning wordt opgesteld door het management (projectmanager). Echter, een vertaling maken van de verbetering van het functioneren van het management betekent concreet niets voor de planning. De focus in deze scriptie ligt op de inhoud en structuur van de planning. Hooguit kan worden opgemerkt dat bij een beter functionerend management een betere kwaliteit planning mag worden verwacht, maar dit verschilt per organisatie. Gezocht wordt naar veranderingen van de planning die niet

afhankelijk zijn van specifieke details van een project, zoals het functioneren van het management. Het is voor de aangedragen verbeteringen van de kwaliteit van een project niet mogelijk om de betreffende verbeteringen te vertalen naar veranderingen in de planning.

Het nut (business value) van een project is, zoals eerder al beschreven in deze scriptie, vaak onduidelijk of niet verwerkt in een project(voorstel). Redenen daarvoor zijn de problemen met het expliciet maken van het nut en de verschillende vormen van nut (business value) die een project kan hebben. Als verbeteringen werden genoemd het expliciet vermelden c.q. uitwerken van de business value in al haar vormen en op een zodanige manier dat deze te meten is (voor, tijdens en na een project). De tweede verbetering was het integreren van de modellering van de business value in het ontwerp van de software.

Voor de eerste verbetering is het moeilijk om een connectie te maken met de planning. Het expliciet maken van de business value kan wel het nut van een project verduidelijken (en daarmee de kwaliteit van het resultaat positief beïnvloeden), maar concreet heeft dit nog geen gevolgen voor de planning. Maar met behulp van de tweede verbetering is dit wel het geval. Als de modellering van de business value wordt meegenomen in het ontwerp en het is expliciet (meetbaar) beschreven, kan er regelmatig bekeken worden wat de business value van het project is. Een concrete verandering van de planning aan de hand van de combinatie van de twee verbeteringen is het toevoegen van (meerdere) momenten waarop de business value gemeten wordt. Dit kan dan worden verwerkt als een 'work product' in de planning. Normaal gebeurt dit (voor)al aan het begin van een project, maar door deze verandering kan dit in het begin, tijdens en na het project worden gedaan. Hierdoor is de business value dus op elk gewenst moment in een project zichtbaar te maken en kan er bijgestuurd worden zodat het resultaat de gewenste business value heeft. Deze verbetering is van toepassing op elk software project. Het management is verantwoordelijk voor elk project en verzorgt het maken van de planning. Deze verbetering moet daarom door het management worden toegepast. Dit betekent dat de voorgestelde verbetering te plaatsen is in het geïntegreerde model onder het onderdeel *Planning/way of controlling*.

5.5 Een alternatief

In de bespreking van de onderwerpen kwaliteit, het nut en de planning is naar voren gekomen dat allen gebruikmaken van indicatoren om te meten hoe het gesteld is met de kwaliteit, het nut of de algemene voortgang van een project. Een concrete verbetering van de planning op het gebied van het nut is in de sectie hiervoor gegeven. Voor de kwaliteit was er volgens het literatuuronderzoek wat hiervoor is gedaan, geen verbetering te noemen van de planning. Echter, dit betekent niet dat alle mogelijkheden voor verbeteringen zijn uitgeput.

Een alternatief voor de, in de vorige sectie genoemde, verbetering is een andere opzet van de planning, waarbij meetmomenten voor de kwaliteit, het nut en de progressie van een project worden meegenomen. De planning werd tot nu toe gebruikt om puur het werk behorende bij de op te leveren software op te delen en te verdelen over de beschikbare tijd. Een verandering hiervan is het integreren van meetmomenten voor de kwaliteit, het nut en de progressie. Dit betekent concreet dat de planning niet uitsluitend uit taken en/of work products zal bestaan die te maken hebben met de op te leveren software (of delen daarvan), maar ook uit taken met betrekking tot de kwaliteit, het nut en de progressie van een project. Dit kan op twee manieren:

1. de taken van een work product uit de planning worden uitgebreid met een taak *kwaliteitscontrole* en *business value assessment*. Dit houdt in dat bij elk work product een controle van de kwaliteit plaatsvindt (meten van waarden van de indicatoren) en de business value in beeld gebracht wordt die het project op het moment bevat. Na deze taken volgt dat een taak *progressie update* die de progressie van het project in beeld brengt
2. *kwaliteitscontrole*, *business value assessment* en *progressie update* vormen elk een work product op zich en worden op geschikte punten in de planning toegevoegd (bijv. aan het begin van een project, tijdens een project op een aantal punten en aan het eind)

Voor beide opties geldt dat ze toegepast moeten worden door het management: immers, het management maakt de planning voor een project. Hiermee valt het alternatief onder het onderdeel *Planning/way of controlling* uit het geïntegreerde model.

5.6 Verbeteringen van de planning voor een willekeurig type IT project

Als er gekeken wordt naar verbeteringen van de planning om de kwaliteit en het nut van projecten (ongeacht het type) positief te beïnvloeden, dan kan de volgende verbetering worden aangebracht: de *progressie update*, *business value assessment* en de *kwaliteitscontrole* moeten worden toegevoegd aan de planning. Om deze onderdelen te meten kan de meetmethode die voorgesteld is in hoofdstuk 3 van deze scriptie worden gebruikt. Voorwaarde voor het gebruik van deze methode is wel dat er invullingen worden gedefinieerd voor de indicatoren en dat er normen worden vastgesteld. Als de verbetering geplaatst wordt binnen het geïntegreerde model, valt dit onder het onderdeel *Planning/way of controlling*.

Door het structureel meten van de kwaliteit, het nut en de progressie is men op elk gewenst moment tijdens een project op de hoogte van de stand van zaken en kan er sneller worden ingegrepen om een (mogelijke) mislukking te voorkomen. Op deze manier kunnen de kwaliteit en het nut van projecten door middel van de planning worden verbeterd.

Chapter 6

Conclusie

Nu alle deelvragen zijn behandeld, is het mogelijk om de onderzoeksvraag van de scriptie te beantwoorden:

Kan een planning bijdragen aan een hogere kwaliteit en een beter nut van een IT-project en zo ja, hoe?

Eerst was het nodig om duidelijk te maken wat er onder een project, de kwaliteit en het nut verstaan wordt

Een project is een instantie van het geïntegreerde model. De activiteit-enindeling van een willekeurig IT project is te vergelijken met de indeling van activiteiten voor een software project: een IT project bevat een activiteit waarin uitgezocht wordt wat precies de bedoeling is van het project, een activiteit waarin de te volgen aanpak wordt beschreven, een activiteit die (delen van) de aanpak visualiseert, een activiteit die deze aanpak uitvoert en tot slot een activiteit waarin het resultaat wordt geïmplementeerd en het project wordt afgesloten.

De kwaliteit van een project kan opgesplitst worden in de kwaliteit van het resultaat van een project (bijv. kwaliteit van software) en de kwaliteit van de uitvoer van een project. Het nut van een project bestaat uit een businesskant en een IT kant. De businesskant is veelal bepalend voor het al dan niet uitvoeren van een project. Om de kwaliteit en het nut te meten kan een methode die worden gebruikt die gebaseerd is op het gebruik van indicatoren. Deze indicatoren geven aan hoe het gesteld is met het project op een bepaald gebied van de kwaliteit of het nut. Er kan een onderscheid gemaakt worden in niet-financiële en financiële indicatoren. Om de methode goed te kunnen gebruiken, is het belangrijk om invullingen te definiëren (bijv.

een vertaling naar onderhoudskosten of kosten van het uitvallen van een systeem) van de indicatoren en een norm vast te stellen. Door regelmatig de waarden van de indicatoren te meten, kan er tijdig worden ingegrepen, als blijkt dat een norm niet gehaald wordt.

Wanneer de kwaliteit en het nut worden geplaatst in het geïntegreerde model, dan valt de kwaliteit en het nut van het resultaat van een project onder de *Planning/way of working* en vallen de activiteiten van het management op het gebied van het waarborgen van de kwaliteit en het nut van een project, onder de *Planning/way of controlling*.

Nu is de onderzoeksvraag te beantwoorden

Om een willeurig type IT project positief te beïnvloeden op het gebied van kwaliteit en het nut, is het mogelijk om 1. het projectmanagement te verbeteren en voor de verbetering van het nut, 2. de business value bewust mee te modelleren in een project. Het management heeft de controle over een project en kan de kwaliteit van het proces en daarmee de resultaten positief beïnvloeden door bijv. lean management toe te passen. Het bewust meemodelleren van de business value in een IT project maakt zichtbaar hoe bepaalde keuzes in een project, invloed hebben op de (uiteindelijke) business value. Op deze manier kan de business value van een IT project positief beïnvloed worden.

Als er gekeken wordt naar verbeteringen van de planning om de kwaliteit en het nut van projecten (ongeacht het type) positief te beïnvloeden, dan kan de volgende verbetering worden aangebracht: de *progressie update*, *business value assessment* en de *kwaliteitscontrole* moeten worden toegevoegd aan de planning. Om deze onderdelen te meten kan de meetmethode die voorgesteld is in hoofdstuk 3 van deze scriptie worden gebruikt. Voorwaarde voor het gebruik van deze methode is wel dat er invullingen worden gedefinieerd voor de indicatoren en dat er normen worden vastgesteld. Als de verbetering geplaatst wordt binnen het geïntegreerde model, valt dit onder het onderdeel *Planning/way of controlling*.

Door het structureel meten van de kwaliteit, het nut en de progressie is men op elk gewenst moment tijdens een project op de hoogte van de stand van zaken en kan er sneller worden ingegrepen om een (mogelijke) mislukking te voorkomen. Op deze manier kunnen de kwaliteit en het nut van projecten door middel van de planning worden verbeterd.

Bibliography

- [1] E. D. Arnheiter and J. Maleyeff. The integration of lean management and six sigma. *The TQM Magazine*, 17(1):5–18, 2005.
- [2] K. Balla, T. Bemelmans, R. Kusters, and J. Trienekens. Quality through managed improvement and measurement (qmim): Towards a phased development and implementation of a quality management system for a software company. *Software Quality Journal*, 9:177–193, 2001.
- [3] H. Boot. Projectwerk is mensenwerk. artikel op Computable.nl, augustus 2008. gelezen januari 2009.
- [4] D. N. Card, T. L. Clark, and R. A. Berg. Improving software quality and productivity. In *onbekend*, Third Conference on Software Quality and Productivity, pages 235–241, Washington D.C. , USA, 1987.
- [5] A. Cimitile and G. Visaggio. Managing software projects by structured project planning. *International Journal of Software Engineering Management*, 7(4):553–584, December 1997.
- [6] L. de Laat. Project zonder aantoonbare waarde is onstuurbaar. artikel op Computable.nl, augustus 2008. gelezen januari 2009.
- [7] T. P. Van der Weide and H. A. Proper. Evorm-a conceptual modeling technique for evolving application domains. *Data and Knowledge Engineering*, 12(3):313–359, May 1994.
- [8] A. J. J. Dick, M. E. C. Hull, and K. Jackson. Specifying process and measuring progress in terms of information state. *Journal of Systems and Software*, 76(3):311–322, June 2005.
- [9] A. Harada, S. Awane, Y. Inoya, O. Ohno, M. Matsushita, S. Kusumoto, and K. Inoue. *Lecture Notes in Computer Science*, volume 3840, chapter Unifying the Software Process Spectrum, pages 249–261. 2005.

- [10] B. Henderson-Sellers and C. Gonzalez-Perez. A comparison of four process metamodels and the creation of a new generic standard. *Information and Software Technology*, 47(1):49–65, January 2005.
- [11] S. Komiya and A. Hazeyama. A meta-model of work structure of software project and a framework for software project management systems. *IEICE Transactions on Information and Systems*, E81D(12):1415–1428, December 1998.
- [12] S. Langendijk. Agile rekent af met falend projectmanagement. artikel op Computable.nl, juli 2008. gelezen januari 2009.
- [13] A. Louws. Agile projectmanagement is kritische succesfactor. artikel op Computable.nl, oktober 2008. gelezen januari 2009.
- [14] R. Madachy. *Integrated Modeling of Business Value and Software Processes*, chapter onbekend, pages 389–402. Springer-Verlag Berlin Heidelberg, 2005.
- [15] P. V. Martins and A. R. da Silva. Pit-p2m: Projectit process and project metamodel. In R. Meersman et al., editor, *Lecture Notes in Computer Science*, volume 3762, pages 516–525. Springer-Verlag Berlin Heidelberg, 2005.
- [16] T. Motoyama. Improving software development through three stages. *IEEE Software*, 23(5):81–87, September-October 2006.
- [17] R. S. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Higher Education, 2003.
- [18] H. A. (Erik) Proper. *A Theory for Conceptual Modelling of Evolving Application Domains*. PhD thesis, Katholieke Universiteit Nijmegen, 1994.
- [19] P. S. Seligmann, H. G. Sol, and G. M. Wijers. Analyzing the structure of i.s. methodologies, an alternative approach. In R. Maes, editor, *onbekend*, Proceedings of the First Dutch Conference on Information Systems, 1989.
- [20] P. Simmons. Quality outcomes: Determining business value. *IEEE Software*, 13(1):25–32, January 1996.
- [21] R. van Heur. Klant is zwakste schakel in ict-projecten. artikel op Computable.nl, juni 2008. gelezen januari 2009.

- [22] R. van Heur. Vijf tips voor ict-projecten bij de overheid. artikel op Computable.nl, augustus 2008. gelezen januari 2009.
- [23] R. van Heur. Zonder business case mislukt ict-project. artikel op Computable.nl, augustus 2008. gelezen januari 2009.

Bijlagen

Chapter 7

Planning

Hieronder is de planning voor mijn bachelorscriptie te vinden. Ik heb de planning verdeeld in 3 delen:

- De globale planning van mijn scriptie, met daarin een schets van de te verrichten activiteiten en de deadline die ik wil aanhouden
- Een specifieke planning van de uitwerking van de deelvragen per dag bekeken (nav gesprek met begeleider op 02-04-2009)
- De afspraken met de begeleider: Bij het onderwerp zal steeds genoemd worden wat precies besproken zal worden

7.1 Globale planning

Activiteit	Tijdsinschatting
afmaken onderzoeksplan en invoeren in werkplaats	1 week (9-13 februari)
uitwerking deelvraag 1	1 week (16-20 februari)
uitwerking deelvraag 2	1 week (23-27 februari)
uitwerking deelvraag 3	1 week (2-6 maart)
uitwerking deelvraag 4	1 week (9-13 maart)
uitloop (uitwerking deelvragen)	1 week (16-20 maart)
analyse van resultaten/verwerken deelvragen	1 week (23-27 maart)
conclusie uitwerken/afmaken	1 week (30 maart-3 april)
afronden scriptie	1 week (6-10 april)
verbeteren/aanpassen scriptie	1 week (13-17 april)
uitloop	2 weken (20-30 april)

De beoogde deadline voor de scriptie is 30 april/1 mei.

7.2 Herziene planning

- vraag 1: wat is een project?
- vraag 2: wat is de kwaliteit en het nut van een project?
- vraag 3: kunnen de kwaliteit en het nut van uitkomsten van een project positief worden beïnvloed?
- vraag 4: hoe kan ervoor gezorgd worden dat een project positief wordt beïnvloed met behulp van de planning?

Deelvraag	Omschrijving	Deadline
vraag 1	software engineering, model standaardonderdelen	8-4-2009
vraag 1	seligmann theorie: uitleg, model	9-4-2009
vraag 1	connectie SE en Seligmann theorie	10-4-2009
vraag 1	resultaatmodel	13-4-2009
vraag 2	kwaliteit van een project uitwerken	14-4-2009
vraag 2	nut van een project uitwerken	15-4-2009
vraag 2	connectie met Seligmann theorie	16-4-2009
vraag 3	mogelijkheden tot verbetering: nut, kwaliteit	17-4-2009
vraag 3	mogelijkheden koppelen aan Seligmann theorie	20-4-2009
vraag 4	plaatsing planning model Seligmann	21-4-2009
vraag 4	herhaling verbeteringen	22-4-2009
vraag 4	beargumenteren of verbeteringen mogelijk zijn voor planning	23-4-2009
vraag 4	beargumenteren of verbeteringen mogelijk zijn voor planning	24-4-2009
conclusie	conclusie uitwerken	27-4-2009
reflectie	nalopen, verbeteren	29-4-2009
afronding	afronding (en pdf maken)	30-4-2009

7.3 Afspraken met de scriptiebegeleider

Datum	Onderwerp	Tijdstip
17-12-2008	algemene gang van zaken/bachelorscriptie	10.00u
07-01-2009	bespreking nieuw onderwerp voor scriptie	15.00u
29-01-2009	bespreking onderzoeksplan (vragen, problemen)	14.30u
10-02-2009	bespreking onderzoeksplan	17.00u
06-03-2009	bespreking voortgang scriptie	10.00u
18-03-2009	bespreking voortgang scriptie	17.00u
02-04-2009	bespreking voortgang scriptie	16.00u

Chapter 8

Logboek

Datum	Activiteit	Uren
17-12-2008	algemene gang van zaken/bachelorscriptie	0,5
05-01-2009	brainstormen over nieuw onderwerp	4
07-01-2009	bespreking nieuw onderwerp voor scriptie	0,5
27-01-2009	uitwerken onderzoeksplan	2
28-01-2009	uitwerken onderzoeksplan	1
29-01-2009	bespreking onderzoeksplan	0,5
09-02-2009	opzet scriptie in digitale werkplaats	5
10-02-2009	bespreking digitale werkplaats/ afmaken onderzoeksplan	0,75
11-02-2009	uitwerken onderzoeksplan in digitale werkplaats	3,75
16-02-2009	zoeken naar literatuur voor uitwerking deelvraag	0,5
17-02-2009	zoeken naar literatuur voor deelvraag 1	7
18-02-2009	theoretisch kader uitgewerkt	1
18-02-2009	inlezen literatuur deelvraag 1	1,75
20-02-2009	verbeteringen aangebracht in onderzoeksplan	2
25-02-2009	inlezen literatuur behorende bij deelvraag 1	2
06-03-2009	bespreking voortgang scriptie	0,5
10-03-2009	zoeken naar literatuur voor deelvraag 2	2,5
16-03-2009	zoeken naar literatuur voor deelvraag 3	0,75
17-03-2009	uitwerken /analyseren gevonden informatie voor deelvraag 1	2,5
18-03-2009	uitwerken deelvraag 1	4,5
18-03-2009	bespreking voortgang scriptie	0,5
02-04-2009	bespreking voortgang scriptie	0,75
08-04-2009	software engineering en model standaardonderdelen afgemaakt	1,75
09-04-2009	Seligmann theorie en model afgemaakt	1,5
10-04-2009	connectie SE-framework en seligmann theorie afgemaakt	2,75
13-04-2009	resultaatmodel afgemaakt	1
15-04-2009	kwaliteit van een project afgemaakt	2,25

Datum	Activiteit	Uren
20-04-2009	nut en connectie theorie/model afgemaakt	4,5
21-04-2009	aanvullende literatuur gezocht voor deelvraag 3	1,5
22-04-2009	mogelijkheden ter verbetering en koppeling met model afgemaakt	5,5
23-04-2009	plaatsing planning in het model, herhaling verbeteringen afgemaakt	3
27-04-2009	beargumenteren of verbeteringen mogelijk zijn voor planning	1,75
28-04-2009	beargumenteren of verbeteringen mogelijk zijn voor planning	1,25
29-04-2009	conclusie uitwerken afgemaakt	3,5
30-04-2009	reflectie (opstart) gemaakt	3
01-05-2009	reflectie: nalopen, verbeteren (1e keer scriptie nagelopen)	3
06-05-2009	reflectie: nalopen, verbeteren (2e keer scriptie nagelopen)	4
07-05-2009	reflectie: nalopen, verbeteren (3e keer scriptie nagelopen)	4
03-06-2009	verbeteren deelvraag 1 nav feedback scriptiebegeleider	5
04-06-2009	verbeteren deelvraag 2 nav feedback scriptiebegeleider	5,25
08-06-2009	verbeteren deelvraag 3 en 4 nav feedback scriptiebegeleider	4
08-06-2009	aanpassen conclusie met behulp van verbeteringen v/d deelvragen	0,5
10-06-2009	omzetten werkplaatsuitwerking van de scriptie in een latex file	9
11-06-2009	omzetten werkplaatsuitwerking van de scriptie in een latex file	5
17-06-2009	omzetten werkplaatsuitwerking van de scriptie in een latex file	4
21-06-2009	verbeteren scriptie (deelvragen 1 en 2)	4
22-06-2009	verbeteren scriptie (deelvragen 3 en 4, conclusie)	4