



Fachhochschule Nordwestschweiz
Hochschule für Technik

SOFTWARE-ENTWICKLUNGSPROZESSE

Aufgabe 2.2

Gegenüberstellung Scrum - DAD

10. November 2019

Dozentin:

Peter Sybille

Studierende:

Eggenschwiler Carlo

Frei Dominik

Frommwiler Dominic

Inniger Marco

Zusammenfassung

Inhaltsverzeichnis

1	Einführung	3
1.1	Was ist Dicipined Agile Delivery?	3
1.1.1	Phasen in DAD	4
1.1.2	DAD als Scrum-basiertes Vorgehensmodell	5
2	Gegenüberstellung	6
2.1	Planung	6
2.1.1	Release-Planung	6
2.1.2	Priorisierung	7
2.1.3	Planungsumfang	7
2.1.4	Planungsaufwand	8
2.1.5	Risikomanagement	8
2.2	Zuständigket und Zusammenarbeit	9
2.2.1	Rollen	9
2.3	Empfehlung	9
3	Gegenüberstellung Zusammenarbeit	10
3.1	Zuständigkeit	10
3.1.1	Koordination im Team	10
3.1.2	Rolle und Aufgaben des Kunden	11
3.2	Empfehlung	12

1. Einführung

Diese Arbeit zeigt auf wie DAD auf Scrum angewendet wird und welchen Mehrwert dadurch gewonnen wird. Wir fokussieren uns explizit in dieser Arbeit nur auf Scrum, da ein Vergleich mit allen agile Methoden nicht umsetzbar gewesen ist. Zudem ist durch den Auftrag gegeben, dass Scrum vom bestehenden Team bereits angewendet wird. Wir empfehlen jedoch als zusätzliche Quelle das Buch «Choose you WoW! A Disciplined Agile Delivery Handbook for Optimizing Your Way of Working».

1.1 Was ist Disciplined Agile Delivery?

Disciplined Agile Delivery (DAD) ist ein Hybrid-Prozess beziehungsweise ein Framework, welches agile Vorgehensmodelle, wie beispielsweise Scrum, integriert. Die Erfinder von DAD (Scott Ambler und Mark Lines) sehen agile Prozesse als nicht voll umfänglich. Sie bieten zusätzliche Fragestellungen und Methoden um die Rahmenbedingungen des Vorgehensmodells zu konkretisieren. DAD ist somit eine Ergänzung zu den Agile Vorgehensweisen wie Scrum, Extreme Programming, Kanban, Lean, etc. Somit ermöglicht DAD bestehende agile Prozesse auf komplexere Unternehmenstrukturen anzuwenden.

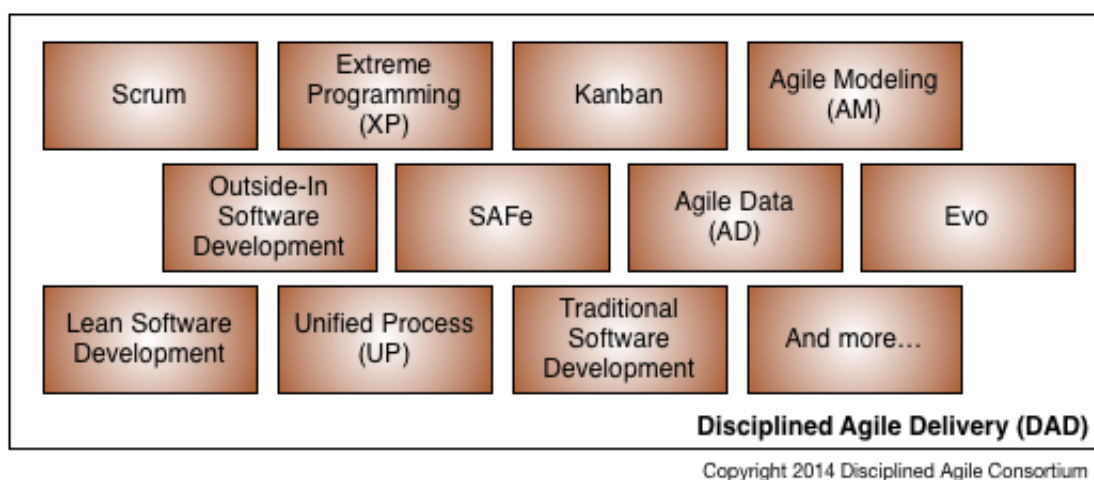


Abbildung 1.1: DAD als hybrides Vorgehensmodell

1.1.1 Phasen in DAD

Allgemein kann gesagt werden das DAD aus drei Phasen besteht Inception, Construction und Transition.

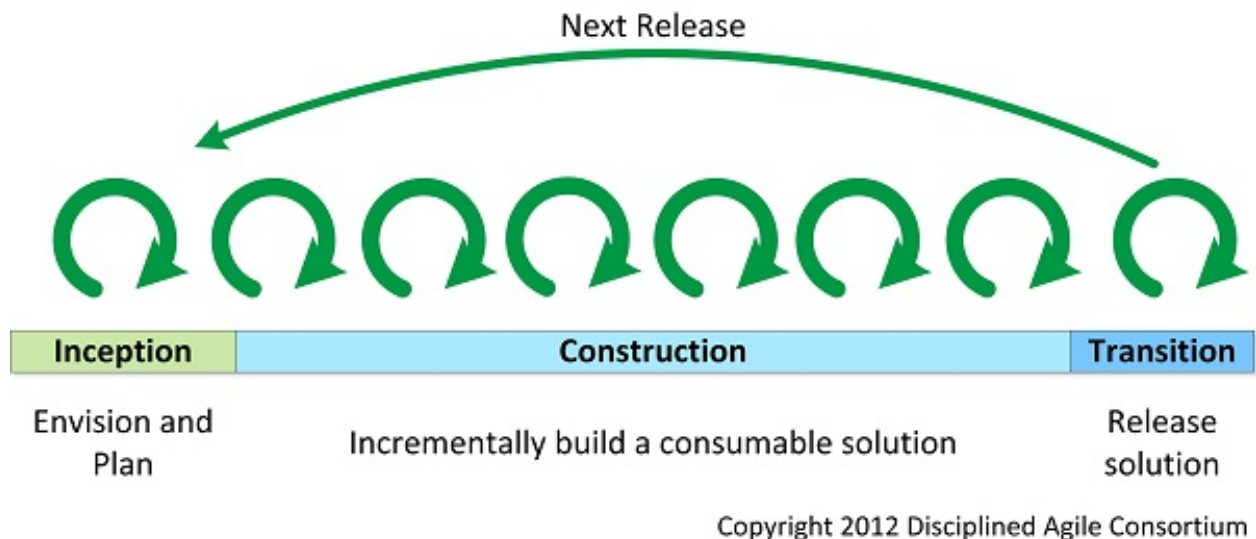


Abbildung 1.2: Highlevel Lifecycle von DAD

In der **Inception** Phase wird die Planung und Analyse des Projekts und dessen Ressourcen gemacht. Dies umfasst folgende Schritte:

- Initiales Team bilden
- Projekt Vision identifizieren
- Mit Stakeholder auf die Projekt Vision einigen
- Auf Unternehmensstrategie abstimmen
- Technische Strategie, initiale Anforderungen und initiale Release-Planung festlegen
- Arbeitsumfeld einrichten
- Finanzierung sichern
- Risiken identifizieren

Die **Construction** Phase beinhaltet die eigentliche Entwicklung und das Testen, wo das entsprechende Vorgehensmodell eingesetzt wird.

- Eine verwendbare Lösung liefern
- Ändernde Bedürfnisse der Stakeholder adressieren
- Näher an das einsetzbare Produkt herankommen
- Qualität verbessern oder auf höhere Qualität erarbeiten

- Architektur frühzeitig beweisen
- Arbeitsumfeld einrichten

Die **Transition** Phase betrifft Zieleinhaltung und Lieferung.

- Einsatzfähigkeit der Lösung sicherstellen
- Empfangsbereitschaft der Stakeholder sicherstellen
- Lösung in produktive Umgebung liefern

1.1.2 DAD als Scrum-basiertes Vorgehensmodell

DAD als agiles Vorgehensmodell bedeutet eine erweiterte Scrum Vorgehnsweise. Scrum wird dort erweitert wo es unzureichend definiert ist.

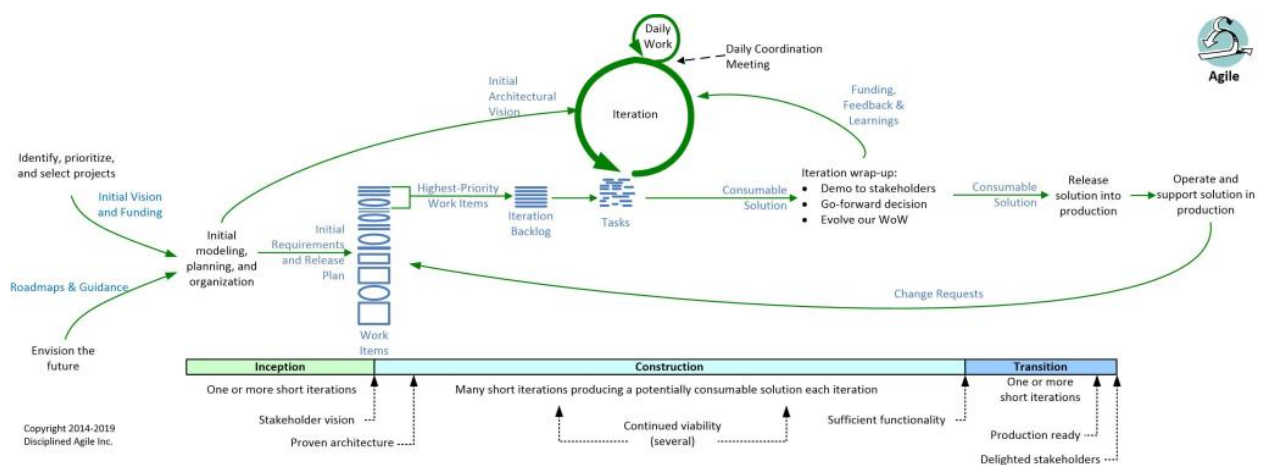


Abbildung 1.3: Scrum-basiertes DAD Vorgehensmodells

Dieses Vorgehensmodell bietet folgende interessante Aspekte:

- Es ist iterationsbasiert.
- Es verwendet keine Scrum-Terminologie.
- Es zeigt Inputs ausserhalb des Vorgehensmodelles an.
- Es gibt eine Workitem-Liste, kein Product Backlog.
- Es enthält explizite Meilensteine.

2. Gegenüberstellung

2.1 Planung

Das Thema Planung wird unter folgenden fünf Aspekten betrachtet:

1. Release-Planung
2. Priorisierung
3. Planungssicherheit
4. Planungsaufwands
5. Nachvollziehbarkeit

2.1.1 Release-Planung

Scrum: [1]

Der Release-Plan ist ein höhergestellter Plan, der mehrere Sprint beinhaltet und während der Release-Planung festgelegt wird. Der Plan definiert welche Features umgesetzt werden und wann diese erfüllt sind. Er dient auch dazu, den Fortschritt innerhalb des Projekts verfolgen zu können. Es können mehrere Releases während des Projekt geplant werden, oder einfach ein finales Release am Ende des Projekts.

Um eine Release-Planung durchführen zu können muss folgendes bekannt sein:

- Ein priorisiertes Scrum-Backlog
- Die Ressourcen des Scrum-Teams
- Zielerfüllungsbedingungen

Ein Release-Plan kann Termin- oder Feature-geführt sein.

Bei Termin-geführten Projekten wird spezifiziert, welche Features bis zu einem bestimmten Termin erfüllt werden können.

Bei Feature-geführten Projekten wird spezifiziert , bis zu welchem Termin das Features erfüllt ist.

Wie der Backlog ist auch der Release-Plan bei Scrum nicht statisch. Dieser kann sich mit dem Backlog ändern oder auch nach jedem Sprint wieder diskutiert und überarbeitet werden.

DAD: [2]

In DAD wird die Release Planung initial in der Inception Phase gemacht. Der Leitfaden empfiehlt für die Releaseplanung folgende sechs Fragen zu beantworten.

- Wer wird an der Planung beteiligt sein?
- Was ist der Umfang unseres Planungsaufwands?
- Was ist unsere Gesamtstrategie, die diesen Plan vorantreibt?
- Wie detailliert sollte unser Plan sein?
- Welche Kadenzen wird das Team annehmen?
- Welchen Ansatz zur Schätzung werden wir wählen?

Damit soll sichergestellt werden, dass grundlegende Managementfragen gegenüber den Stakeholder beantwortet sind. Zudem wird erreicht, dass eine durchführbare Strategie besteht und zwischen Stakeholder und Delivery Team ein gemeinsames Verständnis existiert.

2.1.2 Priorisierung

Scrum: [3]

Das Scrum-Team priorisiert zusammen mit dem Product-Owner die Tasks/Stories aus dem Scrum-Backlog. Wichtig dabei ist, dass nicht nur priorisiert, sondern dass auch sortiert werden muss. Beim Sortieren wird auch die Reihenfolge von Abläufen berücksichtigt. Die Priorisierung geht mit der Sortierung Hand in Hand.

Weiter achtet Scrum auch darauf, dass die wertvollsten Inkremente frühestmöglich umgesetzt werden.

DAD: [4]

Die Priorisierung bei DAD verhält sich ähnlich wie die Release-Planung. Grundsätzlich gilt wieder das Rolling-Wave-Modell. Dass heisst, dass höher priorisierte Features detaillierter spezifiziert werden und tief priorisierte nur grob. Die Priorisierung kann zu jederzeit wieder angepasst werden.

2.1.3 Planungsumfang

Scrum:

In Scrum betrifft der Umfang immer direkt das Produkt. Der Scope beinhaltet Features ausgedrückt z.B. als User Stories. Diese sind Im Scrum-Backlog abgelegt und verwaltet.

DAD: [5]

DAD geht hier einen Schritt weiter und definiert nicht nur Features sondern sogenannte Working-Items. Bei denen werden auch nicht-funktionale Anforderungen definiert wie z.B. Schulungen, Ferien, Unterstützung anderer Teams usw.

2.1.4 Planungsaufwand

Scrum:

Der Planungsaufwand von Scrum ist relativ gering und ist eigentlich im iterierenden Prozess von Scrum bereits integriert. Die Planung wird bei Scrum vor jedem Sprint im sogenannten Sprint Planning gemacht. Dabei werden die zu erledigenden Items definiert. Der Umfang des Sprints wird vom ganzen Team bestätigt.

DAD:

Initial ist der Planungsaufwand bei DAD hoch. Man muss nebst der eigentlichen Planung des Produkts auch diverse Analysen von Ist-Zuständen bezüglich Ressourcen und Zuständen innerhalb des Unternehmens machen um die Rahmenbedingungen für das Projekt zu legen. Während der Construction Phase ist der Planungsaufwand analog jenem von Scrum.

2.1.5 Risikomanagement

Scrum:

Bei Scrum wird das Risikomanagement hauptsächlich durch die Kommunikation zwischen dem Kunden und Team geführt. Dies geht über den Product Owner. Dabei muss der Kunde durch seinen stetigen Einfluss mögliche Risiken ausschliessen können. Er kann dies mittels Akzeptanzkriterien beeinflussen.

Aus Team-interner Sicht ist die Definition of Done das Kontrollinstrument um Qualität aber auch Vollständigkeit sicherzustellen. Als weiteres Instrument dient die Review am Ende jedes Sprints. An jener wird dem Kunde die umgesetzten Items präsentiert. Hier kann der Kunde oder der Product Owner Einfluss bzw. Missverständnisse aufdecken und klären.

DAD:

Um das Risiko von Fehlkommunikation zu verringern werden bei DAD gegenüber Scrum leichte Meilenstein eingeführt, bei denen ein Abgleich mit dem Kunden stattfindet.

Weiter sieht DAD die Planung von festen Releases vor. Damit soll regelmässig Software zur Verfügung gestellt werden um ein Feedback des Kunden zu erhalten und frühzeitig festzustellen, ob man die Anforderungen so erfüllen kann. Dies ist gleich wie bei Scrum.

2.2 Zuständigkeit und Zusammenarbeit

2.2.1 Rollen

Scrum: DAD:

2.3 Empfehlung

Grundsätzlich wäre der Ansatz mit DAD sicher spannend. Ihr Team kann nach wie vor mit Scrum arbeiten und Sie haben die Möglichkeit mit den erwähnten Instrumenten wie leichte Meilensteine, Release Planung der agilen Entwicklung Einfluss zu nehmen und dem ganzen einen ein Rahmen zu geben.

Jedoch wird Sie das Einführen dieses Vorgehensmodell hohen Aufwand kosten. Vor allem muss für das saubere Anwenden von DAD viel Analyse von bestehenden Prozessen und Zuständen innerhalb Ihres Unternehmens betrieben werden.

3. Gegenüberstellung Zusammenarbeit

3.1 Zusammenarbeit

Das Thema Zuständigkeit wird unter folgenden fünf Aspekten betrachtet:

1. Koordination im Team
2. Rolle und Aufgaben des Kunden Da in linearen Vorgehensmodellen der Kunde nur an bestimmten Meilensteinen beteiligt ist, interessiert es zu betrachten welche Aufgaben der Kunde hier hat.

3.1.1 Koordination im Team

Scrum: [1]

Scrum geht grundsätzlich von einer selbstorganisierenden Koordination aus. Einzig der Scrum Master hat eine "koordinierendeRolle, jedoch nur im Bezug auf den Prozess und Schnittstellen ausserhalb des Projekts. Als wichtigstes Instrument für Koordination in Scrum oder auch allgemein in agilen Vorgehensmodellen ist die Kommunikation mit den anderen Team-Mitgliedern. Dies wird in Scrum mit regelmässigen Treffen/Aussprachen realisiert. Wie zum Beispiel das Daily Stand-up oder Scrum of Scrums (Team übergreifend), welche speziell für die Koordination angedacht ist. Weiter Routinen wie Daily Scrum oder Reflektion dienen auch der Koordination und deren Verbesserung, auch wenn das nicht primär ihr Ziel ist.

DAD: [2]

DAD möchte, dass folgende Fragen bezüglich Koordination in einem Scrum-Team geklärt sind um eine effektive Organisation innerhalb des Teams zu haben.

- Wie werden Informationen innerhalb des Teams ausgetauscht?
- Wer darf die vom Team erstellten Artefakte aktualisieren?
- Wie werden wir uns innerhalb des Teams koordinieren?
- Wenn wir Teil eines größeren Teams sind, wie werden wir dann innerhalb dieses Teams koordinieren?

- Wie werden wir mit Enterprise-/IT-Teams wie Enterprise Architects und Data Managern zusammenarbeiten?
- Wie werden wir unsere Release-/Einsatzplanung mit dem Rest des Unternehmens koordinieren?
- Wenn wir geografisch verteilte Teammitglieder haben, wie werden wir dann mit ihnen zusammenarbeiten?

Innerhalb des Teams soll also klar definiert sein, mit welchen Routinen (tägliches Treffen, Video-Konferenzen, usw.) Informationen zwischen den Teammitgliedern ausgetauscht werden und welche Tools dazu verwendet werden. Auch soll die Zuständigkeit geregelt sein wer die finalen Artefakten verwaltet, das es hier keine Überschneidungen oder Unklarheiten gibt.

Weiter müssen gerade bei agilen Teams innerhalb eines Unternehmen auch die Koordination mit anderen Teams/Abteilungen geregelt sein.

Was oft vernachlässigt wird, ist die Koordination mit Teammitgliedern an anderen geografischen Orten. Der Informationsaustausch wird hier anspruchsvoller, da direkte verbale Kommunikation, welche die effektivste ist, nicht möglich ist.

3.1.2 Rolle und Aufgaben des Kunden

Scrum: [3]

Der Kunde soll in Scrum während des ganzen Projektes immer involviert sein. Folgende Möglichkeiten gibt es den Kunden zu involvieren:

- Der Kunde wird zum Initial-Meeting mit einladen.
- Backlog wird zusammen mit dem Kunden verwaltet.
- Der Kunde nimmt auch an Reviews teil, um Arbeiten zu besprechen und als abgeschlossen zu definieren.

Durch den konsequenten Miteinbezug des Kunden wird das Risiko vermindert, dass der Kunde nicht zufrieden ist, da er fortlaufend Einfluss nehmen kann und mit seiner Teilnahme auch frühe Schritte/Arbeiten bestätigt.

DAD: [4]

DAD hat bezüglich dem Kunden eine andere auch radikalere Haltung. In DAD will man den Begriff "Kunden" nicht verwenden, sondern nur Stakeholder. Dieser werden in folgende Gruppen unterteilt:

- Endbenutzer: Personen die das Produkt schlussendlich verwenden.
- Vorstehende: Personen die schlussendlich entscheiden, welches Produkt beschafft wird, Bezahlungen freigeben, usw. werden

- Partner: Unterhalter, Betreiber, Entwickler von externen Systemen, Juristen, usw.
- Interne: Personen innerhalb des Entwicklungsteams und welche technische oder geschäftliche Dienste liefern

Für ein Produkt gibt es nur Stakeholders und es gilt dessen Anforderung genau zu ermitteln und festzulegen. Dazu werden für alle Stakeholder die Bedürfnisse gleichwertig ermittelt und miteinbezogen.

Man will also bewusst ein Produkt, dass alle Stakeholder gleich berücksichtigen und nicht nur den bezahlenden Kunden hauptsächlich priorisieren. Denn nur so wird der abbezahlende Kunde auch ein nachhaltiges und erfolgreiches Produkt erhalten können. Als wichtig wird auch herausgehoben, dass das Projekt im "wirKontext betrachtet wird und nicht im ihr". Es gibt nicht den Kunden und das Entwicklungsteam, sondern das Projekt betrifft alle gleich.

3.2 Empfehlung

Grundsätzlich wäre der Ansatz mit DAD sicher spannend. Ihr Team kann nach wie vor mit Scrum arbeiten und Sie haben die Möglichkeit mit den erwähnten Instrumenten wie leichte Meilensteine, Release Planung der agilen Entwicklung Einfluss zu nehmen und dem ganzen einen Rahmen zu geben.

Jedoch wird Sie das Einführen dieses Vorgehensmodell hohen Aufwand kosten. Vor allem muss für das saubere Anwenden von DAD viel Analyse von bestehenden Prozessen und Zuständen innerhalb Ihres Unternehmens betrieben werden.

Abbildungsverzeichnis

1.1	DAD als hybrides Vorgehensmodell	3
1.2	Highlevel Lifecycle von DAD	4
1.3	Scrum-basiertes DAD Vorgehensmodells	5

Literaturverzeichnis

- [1] Author nicht erwähnt. Scrum release planning. https://www.scrum-institute.org/Release_Planning.php.
- [2] Scott Ambler. Rolling wave release planning for agile delivery teams. <http://disciplinedagiledelivery.com/rolling-wave-release-planning>.
- [3] Author nicht erwähnt. Ordered not prioritized. <https://www.scrum.org/resources/ordered-not-prioritized>.
- [4] Scott Ambler. Agile core practice: Prioritized requirements. <http://agilemodeling.com/essays/prioritizedRequirements.htm>.
- [5] Scott Ambler. Dad lifecycle - agile (scrum based). <http://disciplinedagiledelivery.com/lifecycle/agile-lifecycle/>.