

# Agile und Lean

KOM-ITIL

Sebastian Meisel

15. März 2025

## 1 Einführung in Agile und Lean Projektmanagement

Agile und Lean sind zwei moderne Ansätze im Projektmanagement, die sich darauf konzentrieren, effizientere und effektivere Prozesse zu schaffen. Sie sind besonders in der IT-Branche beliebt, wo Projekte oft komplex und unvorhersehbar sind.

### 1.1 Typische Faktoren für das Scheitern von IT-Projekten

- Unklare Anforderungen
- Mangelnde Kommunikation
- Rigidität

#### 1.1.1 Agile Projektmanagement

**Grundlagen** Agile basiert auf dem Agilen Manifest (2001), das vier Hauptprinzipien hat:

1. Individuen und Interaktionen über Prozesse und Tools
2. Funktionierende Produkte über umfassende Dokumentation
3. Zusammenarbeit mit dem Kunden über Vertragsverhandlung
4. Reagieren auf Veränderung über das Befolgen eines Plans

1. Individuen und Interaktionen über Prozesse und Werkzeuge: Dieses Prinzip betont die Bedeutung der Menschen, die an einem Projekt arbeiten. Es sagt, dass gute Kommunikation und Teamarbeit wichtiger sind als strikte Prozesse oder spezielle Softwaretools.
2. Funktionierende Software über umfassende Dokumentation: Anstatt viel Zeit in die Erstellung detaillierter Dokumente zu investieren, sollte der Fokus auf der Entwicklung einer funktionierenden Software liegen. Das heißt nicht, dass Dokumentation unwichtig ist, sondern dass sie nicht auf Kosten der Softwarequalität gehen sollte.
3. Zusammenarbeit mit dem Kunden über Vertragsverhandlung: Dieses Prinzip legt Wert auf eine enge Zusammenarbeit mit dem Kunden, um sicherzustellen, dass die entwickelte Software wirklich seinen Bedürfnissen entspricht. Anstatt sich nur auf Verträge und Spezifikationen zu konzentrieren, sollte der Dialog mit dem Kunden im Vordergrund stehen.
4. Reagieren auf Veränderung über das Befolgen eines Plans: In der Softwareentwicklung können sich Anforderungen schnell ändern. Dieses Prinzip sagt, dass Teams flexibel und anpassungsfähig sein sollten, anstatt starr einem vorher festgelegten Plan zu folgen.

### Lösung von IT-Problemen durch Agile

- Anpassungsfähigkeit: Agile Methoden wie Scrum ermöglichen es Teams, flexibel auf Änderungen zu reagieren.
- Kommunikation: Durch tägliche Stand-ups und Retrospektiven wird die Kommunikation innerhalb des Teams und mit Stakeholdern gefördert.

#### 1.1.2 Lean Projektmanagement

**Grundlagen** Lean fokussiert sich auf die Eliminierung von Verschwendung (Waste), um den Wert für den Kunden zu maximieren. Die sieben Arten von Verschwendung sind:

1. Überproduktion
2. Wartezeit
3. Transport
4. Überbearbeitung
5. Lagerung
6. Bewegung
7. Herstellung von Fehlern

1. Überproduktion: Das Erstellen von mehr Produkten oder Funktionen, als tatsächlich benötigt werden. Dies führt zu Lagerkosten und macht es schwieriger, auf Änderungen zu reagieren.
2. Wartezeiten: Zeiten, in denen Teammitglieder oder Ressourcen ungenutzt sind, weil sie auf etwas warten müssen, z.B. auf Entscheidungen oder Feedback.
3. Transport: Unnötige Bewegungen von Materialien, Informationen oder Personen. In der Softwareentwicklung könnte dies das häufige Verschieben von Code zwischen verschiedenen Umgebungen ohne klaren Nutzen sein.
4. Überbearbeitung: Mehr Arbeit in ein Produkt oder eine Funktion stecken, als vom Kunden gewünscht oder benötigt wird. Dies verschwendet Zeit und Ressourcen.
5. Lagerhaltung: Das Aufbewahren von ungenutzten oder halbfertigen Produkten, Materialien oder Informationen. In der Softwareentwicklung könnte dies ungenutzter Code oder nicht implementierte Funktionen sein.
6. Bewegung: Unnötige physische oder mentale Bewegungen, die keine Wertschöpfung bieten. Zum Beispiel könnte das ständige Wechseln zwischen verschiedenen Aufgaben (Task-Switching) als Verschwendung angesehen werden.
7. Fehler/Defekte: Arbeit, die aufgrund von Fehlern, Mängeln oder Unklarheiten korrigiert werden muss. Dies führt zu zusätzlichem Aufwand und Verzögerungen.

### Lösung von IT-Problemen durch Lean

- Effizienz: Durch die Identifizierung und Eliminierung von Verschwendung werden Prozesse effizienter.
- Kundenfokus: Lean legt großen Wert darauf, genau das zu produzieren, was der Kunde benötigt und wann er es benötigt.

### 1.2 Gemeinsame Probleme bei der Implementierung

- Dogmatismus: Die strikte Befolgung der Methoden ohne Anpassung an den Kontext kann kontraproduktiv sein.
- Fehlende Anpassung: Beide Methoden müssen an die spezifischen Bedürfnisse des Projekts angepasst werden.

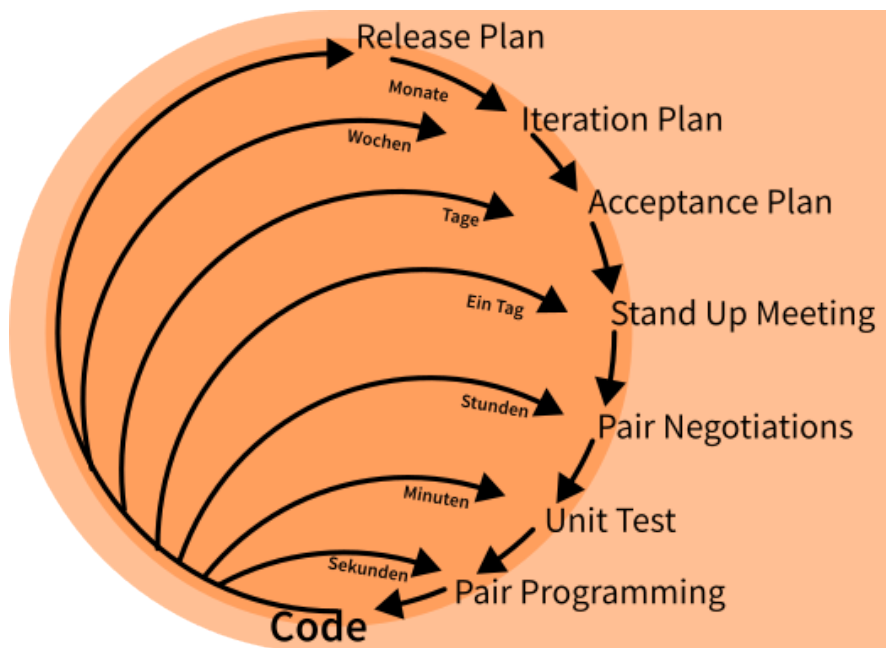
- ***Agile ist mehr auf die Menschen und die Zusammenarbeit fokussiert, während Lean mehr auf Prozesse und Effizienz ausgerichtet ist.***
- ***Beide Ansätze haben ihre eigenen Stärken und Schwächen und können sogar kombiniert werden, um die Vorteile beider zu nutzen.***

## Notes

- Agile und Lean sollten als Werkzeuge betrachtet werden, die je nach Projektanforderungen angepasst werden können.
- Es ist wichtig, die Prinzipien hinter Agile und Lean zu verstehen, um sie effektiv implementieren zu können.
- Sie sollte nicht primär als Mittel zur Zeitersparnis und Produktivitätssteigerung missverstanden werden.

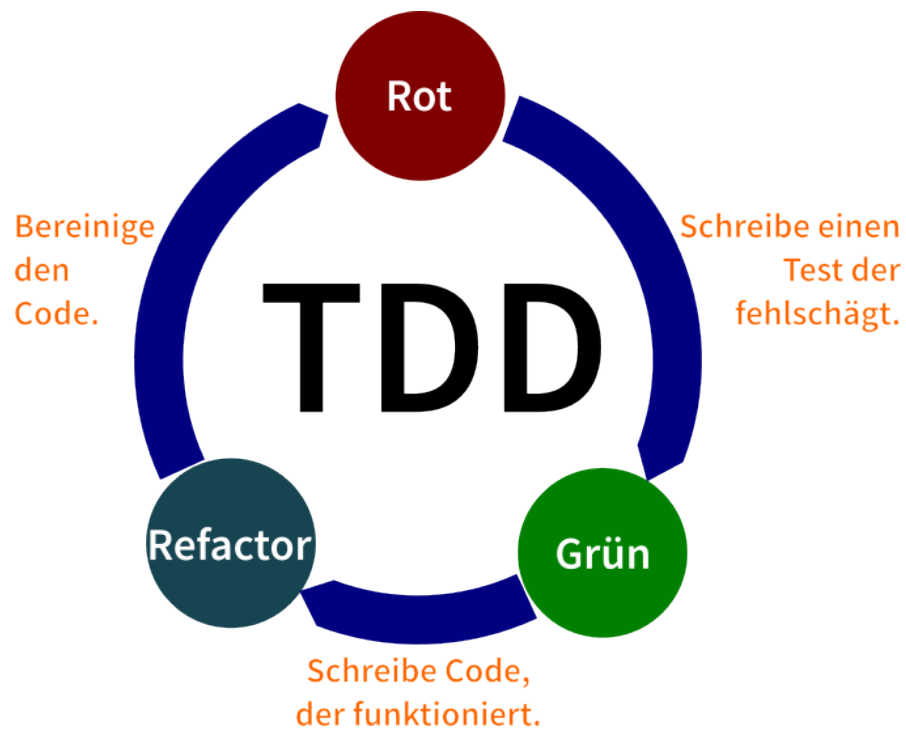
## 2 Agile Methoden

### Extrem Programming (XP)



- Grundlagen Extrem Programming (XP) ist eine agile Methode, die sich auf technische Exzellenz und enge Zusammenarbeit im Team konzentriert.
- Prinzipien
  - Einfaches Design
  - Testgetriebene Entwicklung
  - Pair-Programming
- Weiterführende Ressource Extrem Programming

### Test-Driven Development (TDD)



- Grundlagen TDD ist eine agile Softwareentwicklungsmethode, die das Schreiben von Tests vor dem Code fördert.
- Prinzipien
  - Rot-Grün-Refaktor-Zyklus

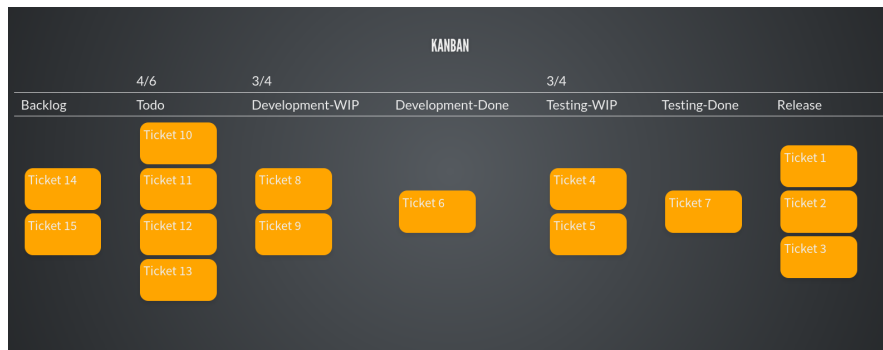
#### Notes

1. Rot: In dieser Phase schreibt der Entwickler einen Test, der die gewünschte Funktionalität beschreibt. Der Test wird fehlschlagen, da die Funktionalität noch nicht implementiert ist. Das Ziel ist es, sicherzustellen, dass der Test korrekt und sinnvoll ist.
2. Grün: Jetzt wird der minimale Code geschrieben, der benötigt wird, um den Test bestehen zu lassen. In dieser Phase ist es nicht wichtig, "schönen" oder effizienten Code zu schreiben; das Ziel ist einfach, den Test "grün" zu bekommen.
3. Refaktor: Nachdem der Test bestanden ist, wird der Code überarbeitet und optimiert, ohne dass sich das Verhalten ändert. Das Ziel ist es, den Code sauberer, effizienter und wartbarer zu machen, während er immer noch alle Tests besteht.

- Weiterführende Ressourcen Test Driven Development: So funktioniert die Methode YouTube: Python Simplified: Python TDD Workflow - Unit Testing Code Example for Beginners (automatisch übersetzte Untertitel verfügbar)

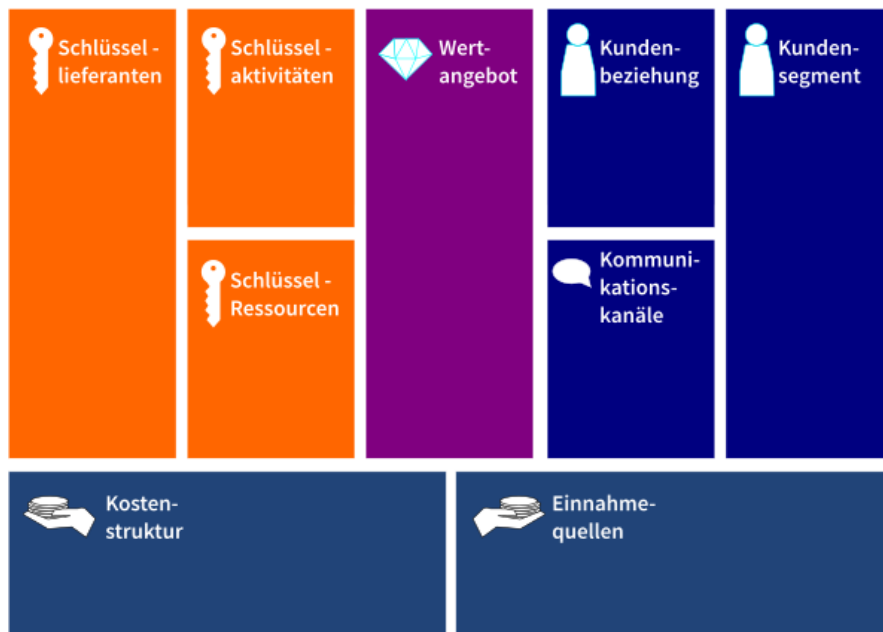
## 3 Lean Methoden

### Kanban



- Grundlagen Kanban ist eine Lean-Methode, die den Arbeitsfluss durch visuelle Darstellung optimiert.
- Prinzipien
  - Visualisierung des Arbeitsflusses
  - Begrenzung der Work-in-Progress-Elemente
- Weiterführende Ressource Was ist Kanban? Ein Einstieg.

## Business Model Canvas



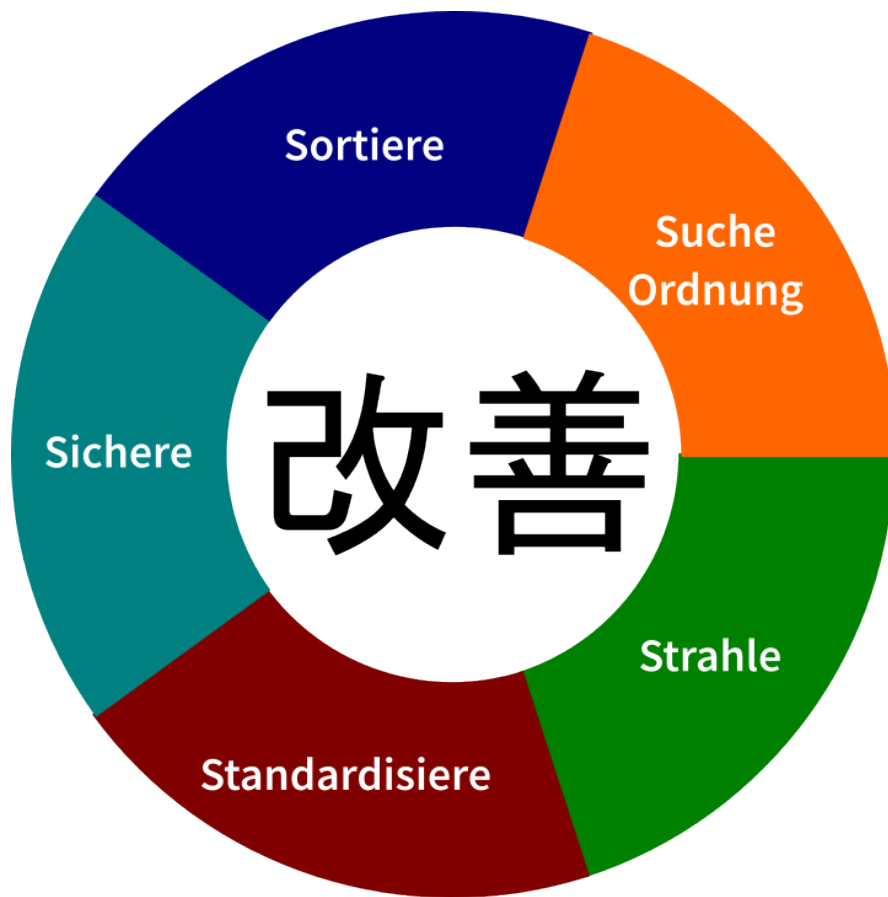
- Grundlagen Das Business Model Canvas ist ein strategisches Lean-Tool zur Entwicklung neuer oder zur Dokumentation bestehender Geschäftsmodelle.
- Prinzipien
  - Neun Bausteine, die die Logik eines Geschäftsmodells darstellen
- Weiterführende Ressource Business Model Canvas: Was es ist und wofür du es brauchst

## Lean Development



- Grundlagen Lean Development bezieht sich auf die Anwendung von Lean-Prinzipien in der Softwareentwicklung.
- Prinzipien
  - Eliminierung von Verschwendung
  - Just-in-Time-Produktion
- Weiterführende Ressource Lean-Softwareentwicklung – Vorläufer oder Nachfolger der Agile-Bewegung?

## Kaizen



- Grundlagen Kaizen ist eine Lean-Philosophie, die kontinuierliche Verbesserung fördert.
- Prinzipien
  - Kleine, inkrementelle Veränderungen
- Weiterführende Ressource Die 5 Grundlagen der Kaizen Methode einfach erklärt