

# Scrum versus Kanban: Welches System passt besser?

Category: Online-Marketing

geschrieben von Tobias Hager | 10. Februar 2026



# Scrum versus Kanban: Welches System passt besser?

Agil ist das neue Schwarz im Projektmanagement – aber was, wenn du zwischen Scrum und Kanban stehst wie ein Entwickler zwischen Stack Overflow und 500 offenen Tabs? Willkommen im Chaos der agilen Methoden! In diesem Artikel zerlegen wir beide Systeme technisch, taktisch und schonungslos – damit du nicht mehr mit Buzzwords um dich wirfst, sondern endlich weißt, welches Framework zu deinem Team, deinem Produkt und deinem Wahnsinnsprojekt passt.

- Was Scrum und Kanban wirklich sind – jenseits der Buzzword-Suppe
- Die technischen Unterschiede zwischen Scrum und Kanban

- Wann du Scrum einsetzen solltest – und wann es dich ausbremst
- Warum Kanban mehr ist als „To-do“, „Doing“, „Done“
- Wie du das richtige System für dein Team auswählst
- Scrum vs. Kanban in der Praxis: Tools, Boards und Metriken
- Schritt-für-Schritt-Anleitung zur Implementierung beider Systeme
- Common Pitfalls: Wo agile Methoden regelmäßig entgleisen
- ScrumBan: Der Frankenstein unter den Frameworks – sinnvoll oder Bullshit?

# Scrum und Kanban erklärt: Agile Methoden ohne Bullshit

Wer über Scrum versus Kanban spricht, sollte erst einmal wissen, was die beiden Frameworks überhaupt leisten – und was nicht. Spoiler: Beide sind Teil des agilen Methodenkoffers, aber sie verfolgen fundamental unterschiedliche Ansätze. Scrum ist ein Framework mit festen Rollen, Events und Regeln. Kanban ist ein Flow-basiertes System ohne starre Zyklen. Beide versprechen bessere Effizienz, schnellere Auslieferung und weniger Chaos. Aber nur, wenn man sie richtig einsetzt – und nicht als Dogma versteht.

Scrum basiert auf Iterationen, sogenannten Sprints. In einem typischen Sprint-Planning wird festgelegt, welche Aufgaben im nächsten Sprint (meist 2 Wochen) erledigt werden. Danach heißt es: Hände weg vom Scope. Der Scrum Master schützt das Team, der Product Owner priorisiert, und das Development Team liefert. Am Ende gibt's eine Sprint-Retrospektive, in der man analysiert, warum wieder alles schiefgelaufen ist – oder auch nicht.

Kanban dagegen sagt: Mach einfach. Aber mach es sichtbar. Und mach nicht zu viel auf einmal. Es gibt keine Sprints, keine festen Rollen, keine Meetings-Marathons. Stattdessen ein Kanban-Board, das zeigt, wo jede Aufgabe gerade steckt – und ein WIP-Limit (Work in Progress), das verhindert, dass du zehn To-dos gleichzeitig anfängst und keins beendest. Das Ziel ist ein kontinuierlicher Fluss (Flow), der Engpässe sichtbar macht und Prozesse automatisch verbessert.

Beide Systeme sind nicht besser oder schlechter – sie sind anders. Scrum ist strukturierter und eignet sich gut für komplexe, inkrementelle Produktentwicklung. Kanban ist flexibler und passt zu Teams, die Service-orientiert arbeiten oder viele Ad-hoc-Aufgaben bearbeiten. Aber der wahre Unterschied liegt in der Denkweise: Scrum will planen. Kanban will fließen lassen.

Das Problem: Viele Unternehmen implementieren „agile“ Methoden, ohne ihre Prozesse, ihre Kultur oder ihre Tools zu verstehen. Das Ergebnis: Scrum wird zur Bürokratie, Kanban zum Kanbanbananaboard, und am Ende schreit jeder nach Wasserfall. Deshalb: Erst verstehen, dann entscheiden. Und dann richtig machen.

# Technische Unterschiede: Scrum versus Kanban im Detail

Scrum und Kanban unterscheiden sich nicht nur im Mindset, sondern auch auf technischer Ebene. Wer glaubt, dass es bei agilen Methoden nur um Boards und Post-its geht, hat das Prinzip nicht verstanden – oder nie wirklich agile Projekte geleitet. Hier kommen die echten Unterschiede, runtergebrochen auf Prozesse, Tools und Metriken.

Erstens: Zeitmanagement. Scrum arbeitet mit Timeboxing. Jeder Sprint hat eine feste Dauer (z. B. 2 Wochen), in der ein definierter Scope abgearbeitet wird. Das bedeutet: Planung, Umsetzung und Review folgen einem klaren Zyklus. Kanban hingegen kennt keine festen Zeiteinheiten. Tasks fließen kontinuierlich durch das System – ohne Sprint-Druck, aber mit Fokus auf Durchlaufzeit (Cycle Time) und Lead Time.

Zweitens: Rollenverteilung. Scrum definiert drei feste Rollen: Product Owner, Scrum Master und Developer. Jede Rolle hat klare Verantwortlichkeiten und ist Teil des Frameworks. Kanban? Hat keine Rollen. Null. Nada. Jeder im Team kann Aufgaben übernehmen, priorisieren oder blockieren – solange die Regeln des Kanban Boards eingehalten werden. Das erfordert Selbstorganisation, aber auch Reife.

Drittens: Metriken und KPIs. Scrum misst Velocity – also die Menge an Story Points, die pro Sprint erledigt wurden. Das funktioniert gut, wenn du vergleichbare Aufgaben und stabile Teams hast. Kanban setzt auf Cycle Time, Lead Time und Cumulative Flow Diagrams. Damit erkennst du Engpässe, Bottlenecks und Durchsatzprobleme – und kannst sie systematisch beheben.

Viertens: Change-Management. In Scrum sind Änderungen am Scope während eines Sprints tabu – alles wird geplant und dann durchgezogen. Kanban ist flexibler: Neue Aufgaben können jederzeit ins System aufgenommen werden, solange die WIP-Limits nicht verletzt werden. Das macht Kanban ideal für Umgebungen mit hoher Volatilität.

Fünftens: Tooling. Scrum lässt sich mit Tools wie Jira, Azure DevOps oder Scrumwise gut abbilden – mit Sprints, Backlogs und Burndown Charts. Kanban funktioniert mit Trello, Kanbanize, Flow-E oder jedem Tool, das Boards mit Spalten und Karten unterstützt. Wichtig ist nicht das Tool, sondern die Disziplin, es sauber zu führen.

## Wann du Scrum brauchst – und wann du es lieber lässt

Scrum ist kein Allheilmittel. Es ist ein Framework für komplexe Produktentwicklung in iterativen Schritten. Das funktioniert super, wenn du ein neues Produkt entwickelst, regelmäßig Feedback brauchst und dein Team aus

motivierten, disziplinierten Leuten besteht. Aber Scrum ist auch anstrengend – und nicht jedes Team ist bereit dafür.

Scrum eignet sich in folgenden Szenarien besonders gut:

- Du entwickelst ein neues Produkt mit unsicherem Scope und hoher Komplexität
- Du brauchst schnelle Feedback-Zyklen und inkrementelle Auslieferung
- Dein Team hat dedizierte Ressourcen und kann sich auf Sprints committen
- Du hast einen Product Owner mit Entscheidungsmacht
- Du willst kontinuierliche Verbesserung durch Retrospektiven etablieren

Scrum ist dagegen fehl am Platz, wenn:

- Dein Team ständig unterbrochen wird und keine Sprint-Fokuszeit hat
- Du keine klaren Rollen oder Verantwortlichkeiten definieren kannst
- Dein Projekt eher wartungs- oder supportlastig ist
- Dein Management „Scrum“ nur als Trendspielzeug betrachtet

In der Praxis sieht man leider oft Fake-Scrum: Daily Stand-ups ohne Sinn, Sprints ohne Review, Product Owner ohne Kompetenz. Das führt nicht zu Agilität, sondern zu Frustration. Wenn du Scrum einsetzt, dann richtig – oder gar nicht.

## Warum Kanban mehr ist als ein Board mit Spalten

Kanban wird oft belächelt: „Das ist doch nur ein Board mit ein paar Spalten, oder?“ Falsch gedacht. Kanban ist ein hochwirksames System zur Visualisierung und Optimierung von Arbeitsprozessen. Es basiert auf Lean-Prinzipien und ist ideal für Umgebungen mit ständig wechselnden Anforderungen, hohem Ticketdurchsatz und Service-Charakter.

Der Kern von Kanban sind sechs Prinzipien:

- Visualisiere den Workflow
- Begrenze die parallele Arbeit (WIP-Limits)
- Manage Flow (Durchsatz optimieren)
- Mach Prozessregeln explizit
- Implementiere Feedback-Schleifen
- Verbessere kontinuierlich (Kaizen)

Kanban zwingt dich, Engpässe sichtbar zu machen. Wenn alle Karten in „Testing“ hängen bleiben, weißt du, wo das Problem liegt. Wenn dein WIP-Limit ständig ignoriert wird, hast du ein Disziplinproblem. Und wenn deine Cycle Times explodieren, brauchst du keine Velocity – du brauchst Analyse.

Kanban eignet sich besonders für:

- Teams mit hohem Ticketdurchsatz (z. B. Support, Ops, Maintenance)
- Prozesse mit unvorhersehbaren Anforderungen

- Organisationen, die nicht auf Scrum-Rituale stehen
- Teams, die kontinuierlich arbeiten statt in Sprints

Aber auch Kanban ist kein Wundermittel. Ohne Disziplin, klare Regeln und saubere Pflege des Boards wird es zur optischen Tapete. Wer Kanban richtig nutzt, kann Prozesse radikal entschlacken – aber nur, wenn er auch bereit ist, regelmäßig hinzuschauen.

# Scrum oder Kanban? So triffst du die richtige Entscheidung

Die Wahl zwischen Scrum und Kanban ist keine Glaubensfrage – sondern eine strategische Entscheidung. Sie hängt ab von Teamstruktur, Projekttyp, Unternehmenskultur und technischer Reife. Wer versucht, Scrum auf ein Incident-Team zu pressen, wird ebenso scheitern wie ein Kanban-Board in einem Feature-getriebenen Produktzyklus.

Hier ein Entscheidungsbaum in Kurzfassung:

- Du hast ein Produktteam mit planbaren Releases? → Scrum
- Du hast ein Service-Team mit wechselnden Aufgaben? → Kanban
- Dein Team ist diszipliniert, aber braucht Flexibilität? → Kanban
- Dein Management will Reports, KPIs und Forecasts? → Scrum
- Du willst beides kombinieren? → ScrumBan (aber mit Vorsicht)

Wichtig: Die Tools sind austauschbar, der Prozess nicht. Ob du Jira, Trello oder Excel nutzt – entscheidend ist, wie du arbeitest. Und ob du die Prinzipien hinter Scrum und Kanban wirklich verstehst – oder nur das Framework auf dein Chaos wirfst, in der Hoffnung, dass es sich selbst sortiert.

# Fazit: Scrum, Kanban – oder einfach mal nachdenken

Scrum oder Kanban – das ist keine Frage von Religion, sondern von Reife. Beide Systeme haben ihre Berechtigung, ihre Stärken und ihre Schwächen. Scrum bringt Struktur, Fokus und Disziplin – aber nur, wenn du die Rollen und Events ernst nimmst. Kanban bringt Transparenz, Flow und Flexibilität – aber nur, wenn du das Board nicht als Deko behandelst.

Wenn du 2025 agil arbeiten willst, reicht es nicht, ein Board aufzuziehen und „Daily“ zu brüllen. Du brauchst ein System, das zu deinem Team passt – nicht eines, das im letzten Buzzword-Webinar genannt wurde. Denk nach, analysiere deinen Kontext – und dann entscheide dich. Aber richtig. Denn Agilität ist kein Framework. Es ist eine Haltung.