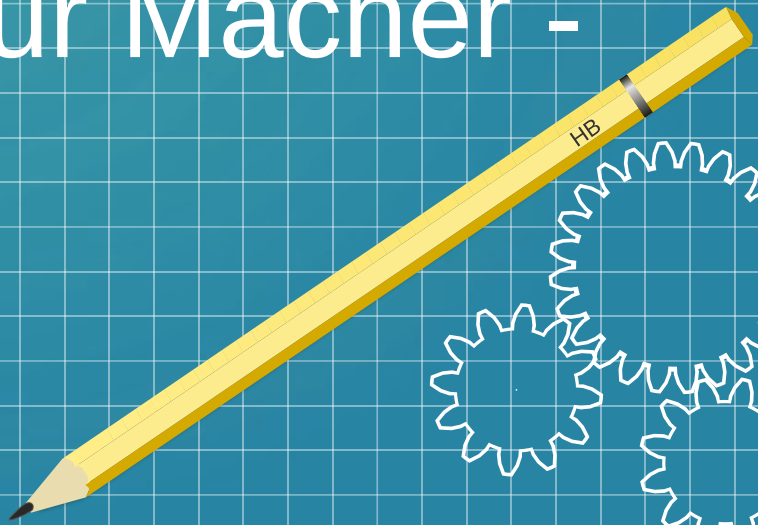
Open Projects, Open Mindset
- Projektmanagement für Macher -



Wer bin ich?

Jürgen Kostzewski

- IT-Projektleiter und
- Open Source-Enthusiast

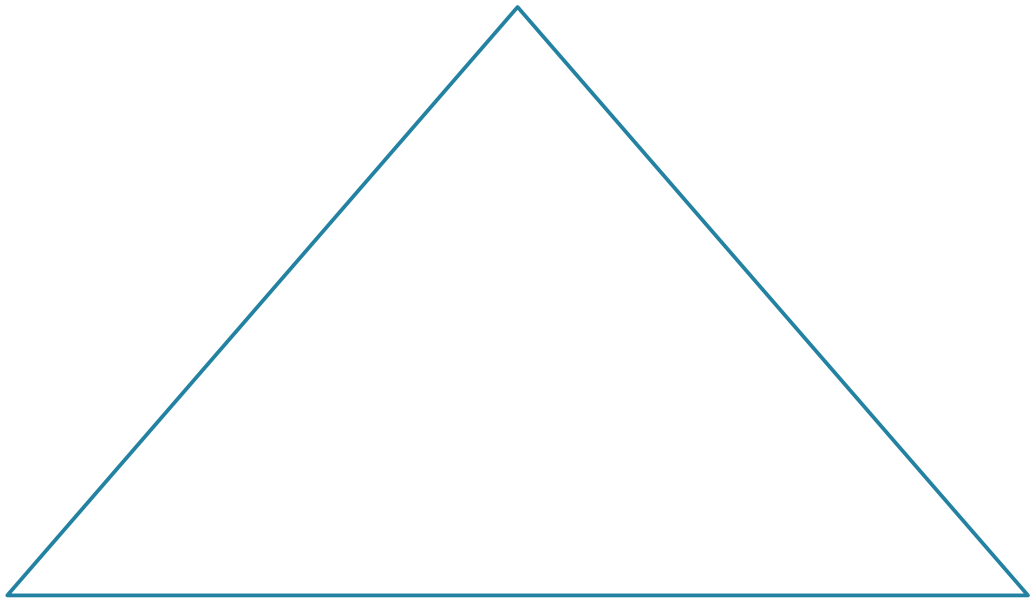
Projekte...

- ...haben ein definiertes **Ziel**.
- ...haben einen definierten **Start** und ein definiertes **Ende**.
- ...haben **begrenzte Ressourcen**.
- ...erfordern häufig **interdisziplinäre Zusammenarbeit**.
- ...haben eine festgelegte **Ergebnisverantwortung**.
- ...sind ein **relativ neuartiges** Vorhaben.
- ...sind häufig **relativ kompliziert** und/oder **komplex**.
- ...sind ein **vom Tagesgeschäft abgetrenntes Vorhaben**.

Zusammengefasst:

Ein Projekt ist
ein einmaliges Vorhaben
mit definiertem Anfang und Ende,
das ein konkretes Ziel
unter Berücksichtigung der Dimensionen
Zeit, Kosten und Qualität
erreichen soll.

Das Magische Dreieck



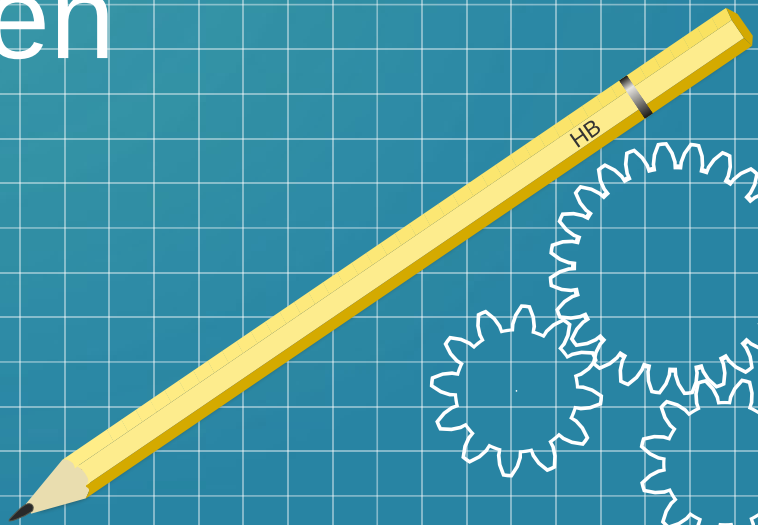
Was ist Projektmanagement?

„Lenken und Leiten eines Projektes im Sinne der Qualität.“

Projekte sind aus technischer Sicht gut, wenn...

- ...es **nachhaltig** betrieben werden kann (Wartbarkeit).
- ...es **gut dokumentiert** ist.
- ...**Vertrauen** in das Ergebnis/Produkt entsteht. (regelmäßige Updates, Roadmap, Prioritäten, etc.)
- ...es einen erkennbaren Nutzen stiftet – ob intern oder öffentlich.

Projektziele und Scope greifbar machen



Wie sollten Projektziele formuliert werden?

- **S**pezifisch
- **M**essbar
- **A**kzeptiert (ethisch und moralisch)
- **R**ealisitisch
- **T**erminiert

Die MoSCoW-Priorisierung

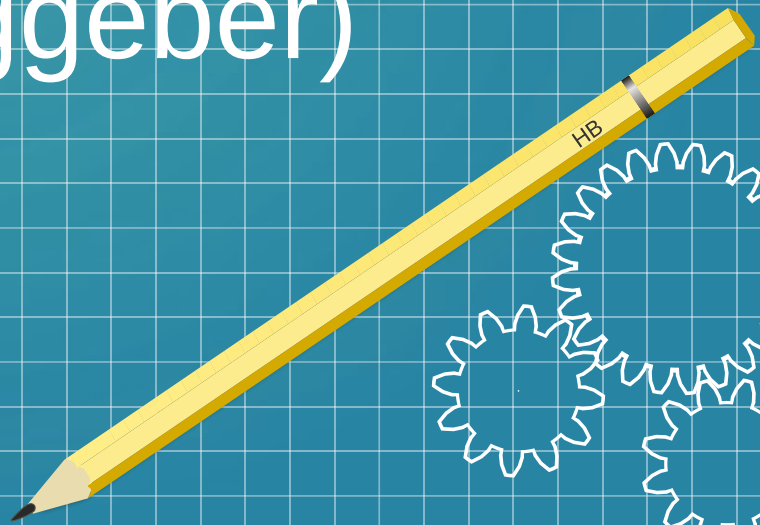
- **M**ust have
- **S**hould have
- **C**ould have
- **W**on`t have (this time) – Out of Scope!

Projektsteckbrief

Ziele des Steckbriefes:

- **Grobe Beschreibung** zu Projektbeginn
- Klärung des **angestrebten Nutzens**
- Erste **Transparenz** über gestellte Anforderungen
- **Formulierung (Klartext)** zu …..mach mal….
- Erste **Zielformulierungen**
- Darstellung wichtiger **Stakeholder**
- Darstellung vorgegebener **Meilensteine**

Stakeholder-Management (auch ohne Auftraggeber)



Wer oder was sind Stakeholder?

„Stakeholder“ sind alle
Personen, Gruppen und/oder Organisationen,
die ein Interesse am Verlauf oder Ergebnis haben
– positiv wie negativ.

Typische Stakeholder in Open-Source-Projekten

- Nutzergruppen
- Community/Maintainer
- Beitragende (Contributors)
- Kritiker
- Sponsoren und Förderer
- Projektteam
- ...

Projekte ohne formalen Auftraggeber

In vielen klassischen Projekt ist klar:

- Es gibt **einen offiziellen Auftraggeber**
- Dieser stellt **Budget** bereit, gibt **Ziele** vor, entscheidet über **Abnahmen**

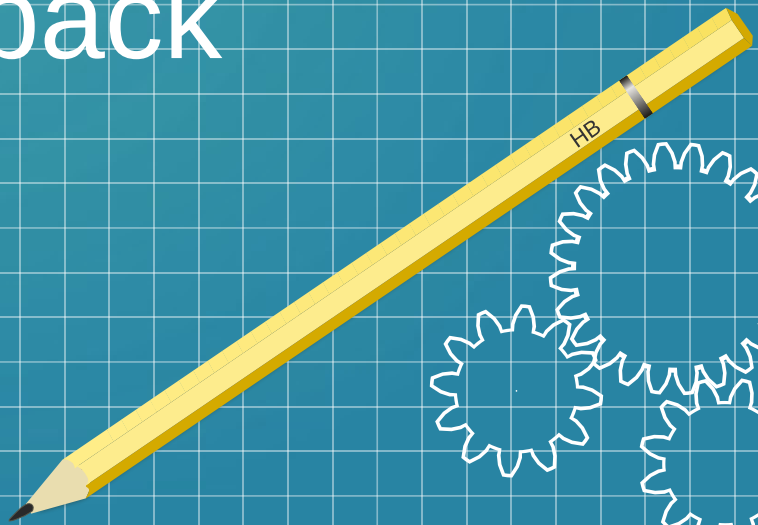
Doch in ehrenamtlichen Open-Source- oder Community-Projekten ist das oft **nicht der Fall**.

Ohne offiziellen Auftraggeber sollten trotzdem…

- …Ziele definiert (und mit anderen abgestimmt) werden.
- …Ressourcen selbst organisiert werden.
- …Vertrauen aufgebaut werden, weil es keine formale Autorität gibt
- …Widerstände früh erkannt werden, weil niemand sie „per Order“ übergehen kann.

Es muss also durch Kommunikation, Motivation, Beteiligung geführt werden – nicht durch Macht!

Aufgaben planen
mit leichtem Gepäck



Typische Herausforderungen

- Zu viel Planung:
  Alles wird in Gantt-Diagrammen erstickt, aber niemand liest es.

- Zu wenig Planung:
  “Wir machen einfach mal“ endet im Chaos oder Stillstand.

Gerade genug Planung, um in Bewegung zu kommen

Fragen zur Selbstreflektion:

- Was ist **unbedingt notwendig**, um anzufangen?
- Was kann sich **im Tun ergeben**?
- Wie erkenne ich, wenn **Nachplanung notwendig** wird?

Tools & Methoden

- Agile Boards
- Backlogs
- Timeboxing

Agile Boards

- Einfaches Pull-basiertes System mit Spalten wie:
  - To Do → In Progress → Done
- Optional: Tags für Prioritäten oder Verantwortliche
- Ideal für Projekte mit unklarer Teamgröße oder fließender Mitarbeit

Nextcloud PLANKA Trello

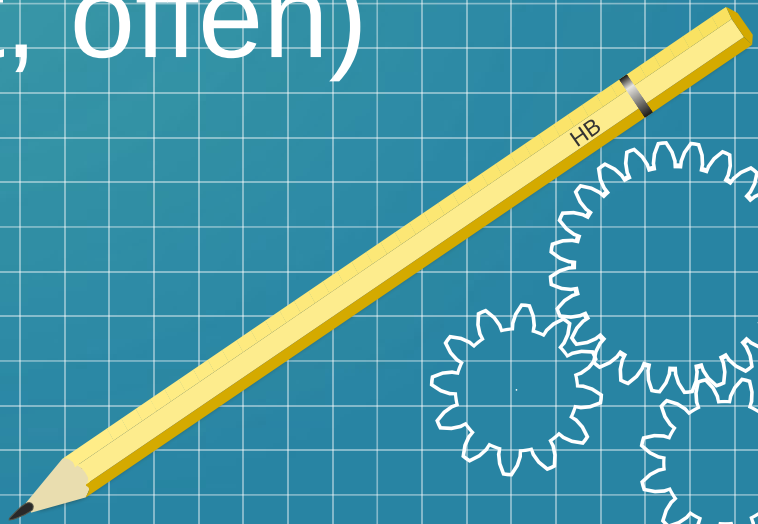
Backlogs

- Aufgabenliste, die nach Wichtigkeit sortiert ist
- Dient als zentraler „Ideenparkplatz“
- Jeder kann Aufgaben vorschlagen (Issues, Tickets, etc.)
- Wichtig: Klarheit über „was rein darf“ (Moderation sinnvoll)

Timeboxing

- Zeitrahmen für Aufgaben und Phasen setzen
- Ziel: Nicht zu viel feilen – sondern machen (Pareto-Prinzip)
- Perfekt für Community-Meetings, Sprints, etc.

Kommunikation in Projekten (asynchron, verteilt, offen)



Warum gute Kommunikation so entscheidend ist

In verteilten, offenen Projekten
ist Kommunikation keine Begleiterscheinung
– sie ist das Betriebssystem.

Typische Herausforderungen

- Niemand weiß, woran die anderen gerade arbeiten.
- Diskussionen versanden im Chat – Entscheidungen sind nirgends dokumentiert.
- Pull Requests ohne Kontext oder Motivation.
- Menschen fühlen sich übergangen oder ausgeschlossen.

Drei kommunikative Prinzipien für verteilte Projekte

1. Asynchron bevorzugen – synchron nur bei Bedarf

- E-Mails, Issues, Wikis, Foren
  → Alle können mitlesen & reagieren, wenn es passt.
  → Weniger Chaos als in Chat-Dauerstreams (z.B. Telegram, etc.).

2. Single Source of Truth

- Jede Info hat **einen klaren Ort**, an dem sie gepflegt wird.

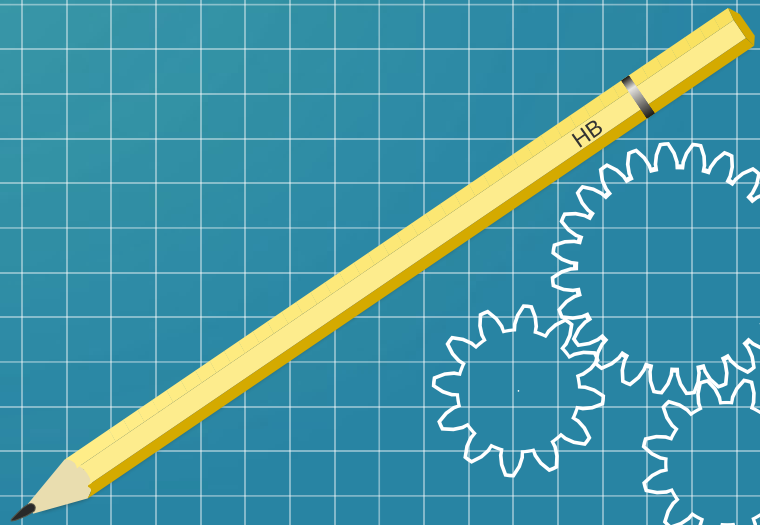
3. Offen & inklusiv kommunizieren

- Nicht jeder ist täglich aktiv
  → Wichtig: Gute Zusammenfassungen, Verlinkungen, klarer Kontext
- Begriffe erklären (Glossar), Abkürzen möglichst vermeiden
- „Fragen sind erlaubt – auf von Neulingen!“

Tools & Formate: Was passt wann?

Ziel	Format/Tool	Bemerkung
Entscheidung dokumentieren	Wiki, Markdown-Datei	Klar: Wer, Was, Warum
Diskussion führen	GitHub Discussions, Forum	Kontext sichtbar halten
Kurze Info teilen	Matrix/Element, Mattermost, Chat-Kanal	Chat nur mit Disziplin!
Status kommunizieren	Kanban-Board	Weniger Rückfragen, mehr Klarheit
Doku bereitstellen	BookStack, Git-Wiki, Nextcloud-Docs	Muss auffindbar & wartbar sein

Change-Management und Motivation



Warum Change-Management in Projekten?

Viele Projekte scheitern nicht an der Technik, sondern an mangelnder Akzeptanz, Veränderungsträgheit oder fehlender Motivation.

Das ist ganz besonders der Fall, wenn...

- ...die Beteiligten freiwillig agieren.
- ...es keine klare Hierarchie gibt.
- ...Vertrauen erst aufgebaut werden muss.

Drei zentrale Fragen im Change-Kontext

1. **Was verändert sich konkret für wen?**
   - Neue Tools, neue Prozesse, andere Zuständigkeiten?
2. **Was verlieren die Betroffenen – oder glauben, zu verlieren?**
   - Status, Kontrolle, Gewohnheit, Bequemlichkeit?
3. **Was ist der Gewinn – und wird er wahrgenommen?**
   - Vereinfachung, mehr Transparenz?

Motivation aktiv fördern I/II

- **Früh beteiligen**
  - „Wir brauchen Mitdenker:innen – hier ist unsere Idee“
- **Wertschätzung zeigen**
  - Release Notes mit Contributor-Nennung, Danksagung
- **Sichtbare Quick Wins erzeugen**
  - „Schaut mal: Der neue (Work-)Flow spart 10 Minuten“

Motivation aktiv fördern II/II

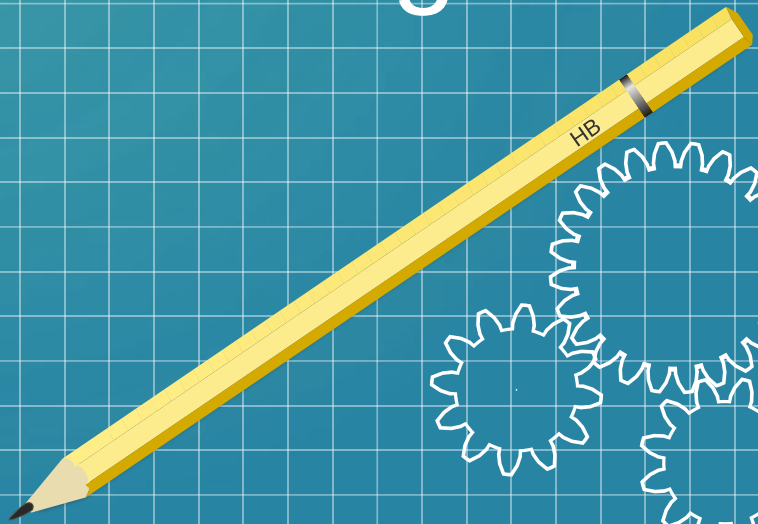
- **Nicht belehren - begleiten**
  - Offene Demos, Fragestunden, einfache Anleitungen
- **Freiwilligkeit ermöglichen**
  - Nicht alles sofort umstellen – Parallelbetrieb denken
- **Rituale & Identifikation schaffen**
  - z.B. Retro, „Projekt-Logbuch“, regelmäßige Updates

Warum Change-Management in Projekten?

„Menschen ändern sich nicht, weil sie müssen – sondern weil sie wollen.

Und sie wollen, wenn sie den Sinn sehen und/oder sich beteiligt fühlen.“

Lessons Learned und Nachhaltigkeit



Warum dieses Kapitel?

In vielen Projekten wird das Ziel irgendwann erreicht – oder das Team verliert den Fokus. Dann passiert oft Folgendes:

- **Das Projekt stirbt still und leise**, ohne dass man daraus lernt.
- **Das Wissen verschwindet mit den Beteiligten** – alles muss später neu gedacht (reengineert) werden.

Gerade in Community- oder ehrenamtlichen Projekten (z.B. Open-Source, Events, Toos) ist das häufig der Todesstoß für die Weiterentwicklung.

Lessons Learned – wie geht das konkret?

1. **Retrospektiven / Projekt-Reviews**
   - Was lief gut? Was hätte besser sein können?
   - Welche Tools/Formate haben sich bewährt?
   - Was war überflüssig?
2. **„Projekt-Logbuch“ oder Abschlusseite/-bericht**
   - Übersicht: Ziele, Umstzung Erkenntnisse, Links
   - Was wurde (nicht) umgesetzt – und warum?
   - Wo sind die Ergebnisse (Code, Docs, Learnings)?
3. **Übergabe ermöglichen**
   - „Wenn jemand das Projekt übernehmen will: Was müsste er wissen?“
   - Offenheit für Forks, Fortsetzungen oder spätere Wiederaufnahme
   - Ggf. Übergabe an anderer Maintainer / Gruppen

Vielen Dank für Eure Aufmerksamkeit