

LIVE-WEBINAR · 19.08.2026

KI in der Softwareentwicklung

**Wie IT-Teams jetzt wirklich entlastet werden:
Der Fahrplan in 4 Schritten**

Live-Webinar · ~45 Min + Q&A

Tobias Morf (Gründer & Inhaber, OPTEN AG)

HOUSEKEEPING

Drei Dinge sind für heute wichtig:

1

Fragen jederzeit in den Chat. Gerne jetzt einmal reinschreiben, woher Sie zugeschaltet sind und wie viele IT-Projekte parallel auf Ihrem Tisch liegen.

2

Aufzeichnung wird im Nachgang per E-Mail zugeschickt.

3

Ablauf: Rund 45 Minuten Input, danach Q&A. Ich versuche mich an die Zeit zu halten.

DAS VERSPRECHEN

In den nächsten 45 Minuten zeigen wir Ihnen, wie Sie KI wirklich in Ihre Softwareentwicklung bringen. Vom Prototyp bis zum Produktivsystem, ohne dass Qualität und Überblick verloren gehen.

Keine Zauberformel. Eine Systematik.

Wir zeigen eine Pipeline, die komplett auf KI umgestellt ist.

Zwischen zwei Extremen.

Die Erwartung an KI in der Entwicklung war nie höher.
Doch in der Praxis scheitert sie öfter, als LinkedIn-Videos zeigen.

DIE LAGE IN DER IT

Beide Wege fühlen sich falsch an.

RISIKO**Extrem 1: Blind drauflos**

Vibe-Coding ohne Struktur.
Schnelle Demos, begeisterte
Chefs und drei Monate später
versteht niemand mehr den Code.
Ownership weg, Qualität weg.

STILLSTAND**Extrem 2: Abwarten & blocken**

Die IT wartet, bis «das Thema reif ist». Offiziell. Inoffiziell prompten die Entwickler längst privat ohne Leitplanken. Und der Backlog wächst.

ZIEL**Der dritte Weg?**

KI-Entwicklung mit System:
schnelle Prototypen UND
sauberer Code UND ein Team, das
es beherrscht. Genau dafür gibt
es den Fahrplan heute.

WAS KI VERSPRICHT

Das Versprechen von KI in der Theorie.

1

Apps in Minuten. Ein Prompt, und die Anwendung steht.

2

Jeder Entwickler wird zum 10X-Entwickler.

3

Gleiches Team, doppelter Output, der Backlog löst sich von selbst.

DIE REALITÄT

Und was wirklich in der Praxis passiert.

- Der Prototyp begeistert aber in der Nutzung zeigen sich UI-Fehler, Overflows, kaputte Umbrüche.
- Dead Code, fehlende Doku, keine Ownership:
Nach drei Iterationen versteht niemand mehr, was da passiert.
- Der CEO fragt, warum es nicht schneller geht.
Der Backlog: unverändert bei 24 Projekten.

WARUM DIE MEISTEN SCHEITERN

«*Wir machen jetzt auch KI-Coding.*»
Und dann? Nichts.

Eine Woche später nachgefragt:

Es ist nichts passiert, ausser drei hübschen Demos.

**Das Problem ist nicht die KI.
Das Problem ist fehlende Systematik.**

DER IST-ZUSTAND IN VIELEN IT-TEAMS

Mehr Tools. Mehr Inseln. Mehr Schatten-IT.

Privates ChatGPT

Copilot-Lizenz

Cursor-Testlauf

Lovable-Demo

v0-Experimente

Claude im Browser

n8n-Insel der IT

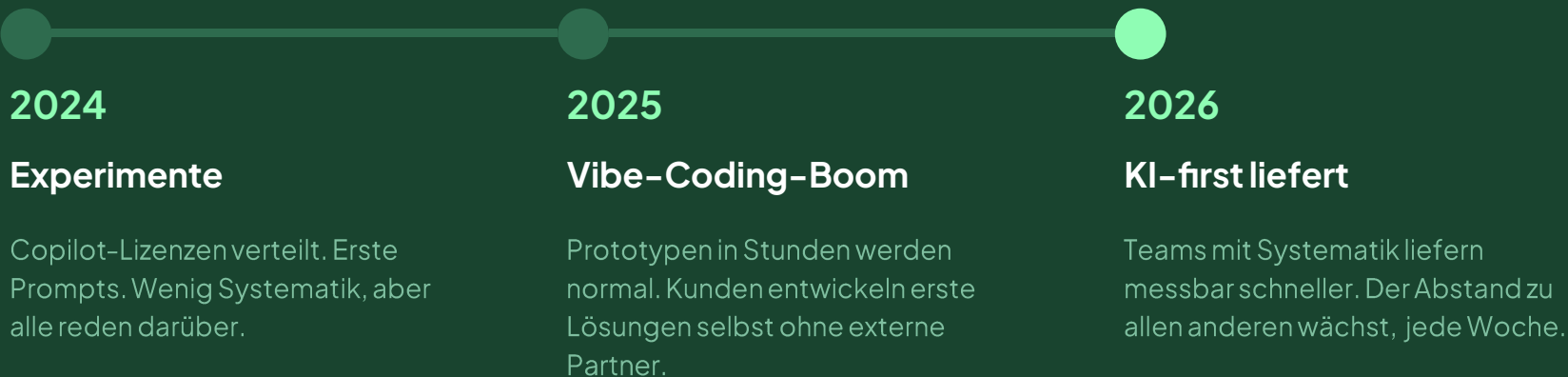
KI-Skript vom
Praktikanten

Viele Inseln. Kaum Überblick. Und jede Woche kommt ein neues Tool dazu.

**«Das grösste Risiko ist nicht die KI selbst,
sondern Code im Unternehmen,
den niemand mehr beherrscht.»**

DER MARKT WARTET NICHT. IN ZWEI MINUTEN, DANN SCHLUSS DAMIT

Die Uhr läuft.



*Und das betrifft Sie schon heute: Ihre Wettbewerber prompten bereits.
Die Frage ist nur, wer es systematisch macht.*

DIE KÖNIGSDISZIPLINEN VON KI

Drei Bereiche mit dem grössten Hebel.

Prototyping

Vertikale Prototypen mit
Echtdaten in Stunden · Konzepte
validieren, bevor budgetiert wird ·
Der No-Brainer-Einstieg.

Testing & QA

Testfälle generieren ·
Regressionstests automatisieren ·
Anspruchsvoller als Prototyping,
aber viel einfacher als früher.

Doku & Refactoring

Legacy-Code verstehen und
dokumentieren · Technische
Schulden sichtbar machen ·
Wissen aus Köpfen in Systeme
bringen.

Es gibt mehr Bereiche. Aber hier entsteht in der Entwicklung am schnellsten sichtbarer Nutzen.

WER DA EIGENTLICH SPRICHT

Aus der Praxis. Nicht aus dem Labor.

17 Jahre

Individualsoftware, Plattformen & Applikationen für den Schweizer Markt – von KMU bis Konzern.

300+ Kunden

Digitalisieren mit OPTEN
u.a. STIHL, Mercedes-Benz,
National League, Gehrig Group.

KI-first

Wir arbeiten selbst täglich mit KI in der Entwicklung. Wir zeigen heute, was wir selbst täglich tun.

SCHWEIZER UNTERNEHMEN

Gleiche Herausforderungen. Immer wieder.

STIHL®

M E R B A G



GEHRIG 80 Jahre
+



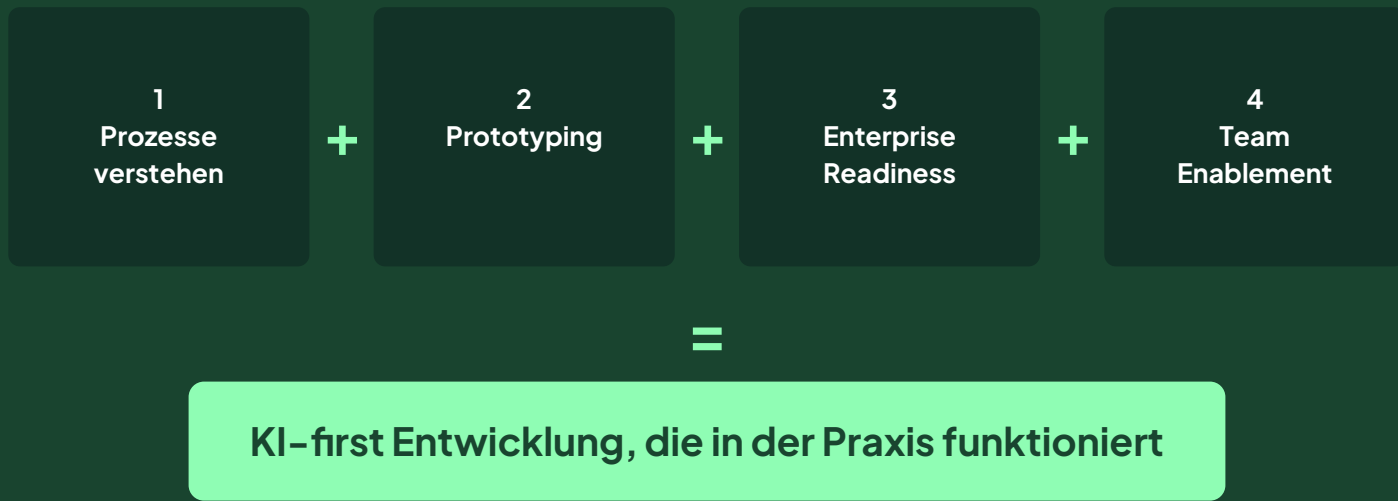
BERNINA
made to create

KEIN THEORETISCHES MODELL. PRAXISERPROBT.

Daraus entstanden: Der 4-Schritte-Fahrplan.

DIE FORMEL

Vier Schritte. Eine Reihenfolge.





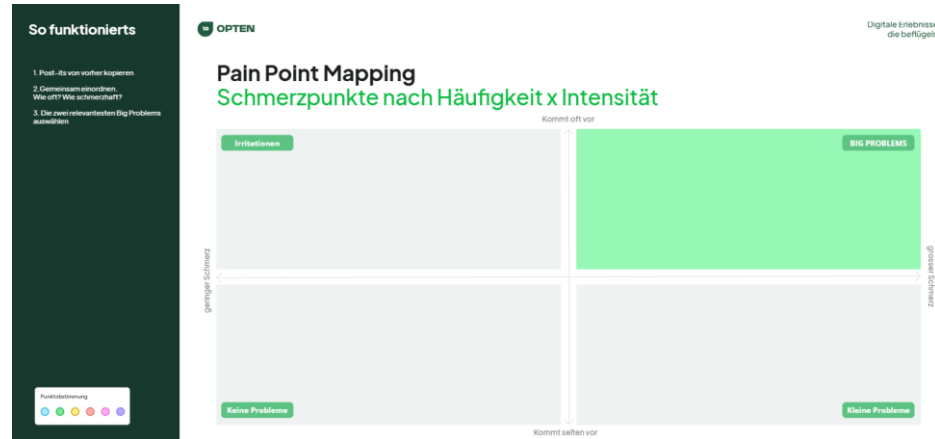
SCHRITT 1 VON 4

Prozesse verstehen

Pain Point Mapping: erst die Potenziale, dann der Detailprozess.

WORKSHOP 1 · POTENZIALE IDENTIFIZIEREN

Pain Point Mapping: wo überhaupt ansetzen?



Auszug aus OPTENs Discovery-Workshop: Unternehmens-Snapshot · Pain Point Mapping · Engpass · Digitalisierungsradar.

Erst der Überblick über die Potenziale. Nur weil ein Prozess digitalisiert ist, ist er noch nicht gut.

WORKSHOP 2 · DETAILPROZESS ANALYSIEREN

Ein Prozess, sauber erfasst.

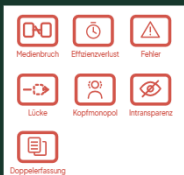
So funktioniert

1. Prozess gemeinsam aufzeichnen
2. Schwachstellen markieren

Wie sieht der Prozess aktuell aus?

"Ob nachgefasst wird, hängt davon ab, wer daran denkt"

Prozessschritt (Was)	Wer	Womit	Wie	Daten
Lead geht ein (Website-Formular, Messe, Anruf)	Marketing / Empfang	Outlook, Website-Formular, Notizzettel	Mailversand	Name, Firma, Anliegen
Lead wird erfasst	Marketing	Excel 'Leadliste' im Sharepoint	manuell abgetippt	Name, Firma, Anliegen
Lead wird Verkäufer zugeteilt	Verkaufsleiter	Mail	Mail-Weiterleitung, persönliche Mitteilung	Zuständigkeit
Erstkontakt durch Verkäufer	Verkäufer	Telefon / Outlook / OneNote	OneNote	Gesprächsnotizen
Anfrage wird besprochen / versendet	Verkäufer	Word / Outlook	Mailversand	Anfrage als PDF
Nachfassen	Verkäufer	Individuelle Reminder	Wenn daran gedacht wird	Keine / private Outlook Reminder
Kunde meldet sich (oder nicht)	Verkäufer	Outlook / Telefon	Mail / Telefonanruf	Zusage / Absage



Detailanalyse eines einzelnen Prozesses (anonymisiert): Was wird heute manuell getan · In welchem System · Dauer · Häufigkeit.

Erst wenn dieser Detailprozess steht, wird gebaut. Nicht vorher.

Ergebnis Schritt 1

**Sie wissen jetzt WO ansetzen.
Jetzt klären wir WIE umsetzen.**

2

SCHRITT 2 VON 4

Prototyping

In Stunden validieren statt monatelang spezifizieren.

RECHENBEISPIEL

Use Case mit grösstem Impact: Prototyping

3 Wochen

Konzeptphase heute: Spezifikation, Abstimmung, Mockups
Bevor eine Zeile Code existiert

→ 2 Tage

Mit KI-Prototyping: klickbarer, vertikaler Prototyp mit Echtdaten

× 6

Projekte pro Jahr, bei denen die Konzeptphase so läuft

≈ 80 eingesparte Personentage pro Jahr

Starten Sie mit EINEM Use Case.

Nicht zehn. Der erste sichtbare Erfolg trägt alle weiteren.

DAS WERKZEUG FÜR SCHRITT 2

Der vertikale Prototyp mit Echtdate.

Kein Mockup, kein Klick-Dummy:

Ein funktionierender Durchstich durch alle Schichten mit Ihren echten Daten.

OPTIMALES SETUP

Unsere Empfehlung für das Prototyping



Installieren, Repository anlegen, loslegen. Reines Vibe Coding.

Im Prototyp ist Wegwerfcode das Ziel. Genau deshalb ist hier Vibe Coding richtig.

SO LÄUFT DAS IN DER PRAXIS

Entwicklungsprozess: Task rein, Code raus.

1 Mensch schreibt Anforderungen

Aufgaben landen als Tasks auf GitHub. Klar beschrieben. Das ist die eigentliche Denkarbeit.

2 Agenten arbeiten parallel

Mehrere Claude-Code-Instanzen ziehen sich die Tasks und arbeiten sie gleichzeitig in separaten Branches und Worktrees ab.

3 Pull Requests zurück

Jede Instanz liefert Code, merged und released diese automatisch. Der Mensch entscheidet **nicht**, was gemergt wird.

4 Manuelle Validierung

Resultat wird manuelle geprüft und Änderungswünsche wiederum als Issue erfasst.

So werden Prototypen gebaut: Task erfassen, Agenten laufen lassen, Ergebnisse prüfen.

WAS DER PROTOTYP BEANTWORTET

Vier Fragen. Vier Antworten.

- 1 Funktioniert der Prozess: Von der Eingabe bis zum Ergebnis, mit Echtdaten?
- 2 Kommt im finalen System KI ins Spiel, wenn ja, wo braucht es (noch) den Menschen?
- 3 Welche Systeme sind betroffen: ERP, CRM, Schnittstellen, Datenbanken?
- 4 Lohnt sich die Umsetzung: Jetzt mit belastbarer Grundlage statt Bauchgefühl?

Email Response Generator

Draft replies from incoming customer emails.

[View history](#)

Agent Configuration

Default Agent

Incoming Email

From

test@marketing.ch

Subject

Kontaktformular verschickt keine Mails mehr

Email Body

Ich habe es selbst getestet: Nach dem Absenden erscheint die Danke-Seite ganz normal, aber im Postfach info@ kommt nichts an, auch nicht im Spam. Geändert haben wir an der Seite meines Wissens nichts.

Können Sie sich das bitte anschauen? Wir bekommen einen grossen Teil unserer Anfragen über dieses Formular.

Freundliche Grüsse
Beatrice Küng
Leitung Marketing

Generate Response

Proposed Response

Guten Tag Frau Küng,

vielen Dank für Ihre Nachricht. Es tut mir leid zu hören, dass Sie seit letzter Woche keine Anfragen über das Kontaktformular erhalten. Ich verstehe, wie wichtig diese Anfragen für Ihr Unternehmen sind.

Um das Problem zu beheben, empfehle ich, die folgenden Schritte zu überprüfen:

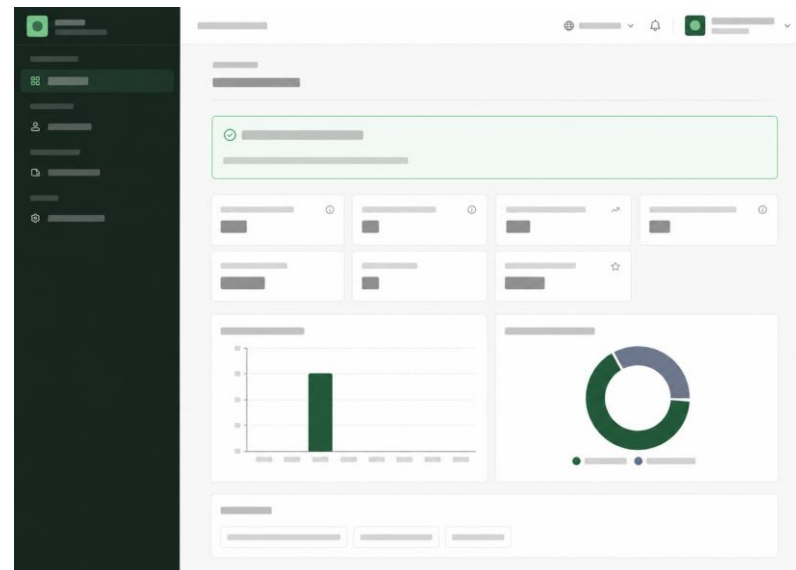
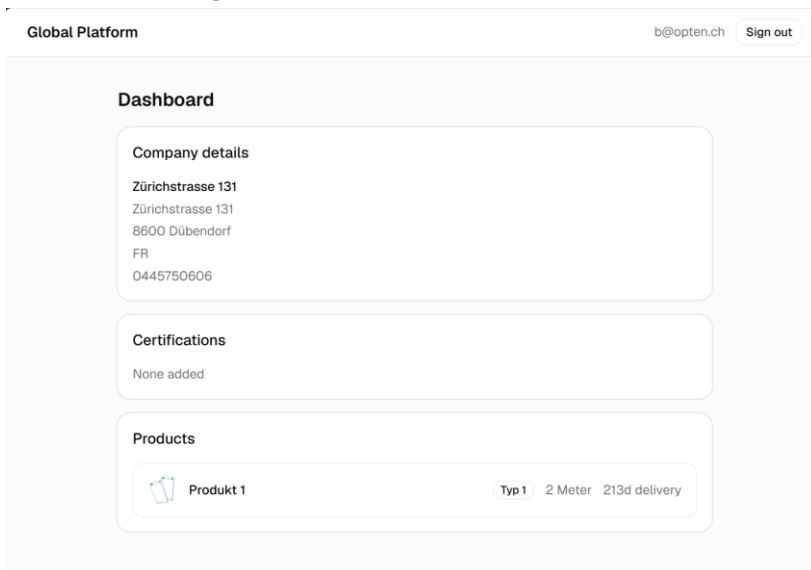
1. **Formularkonfiguration:** Stellen Sie sicher, dass das Kontaktformular korrekt konfiguriert ist und die E-Mail-Adresse für die Zustellung nicht versehentlich geändert wurde.
2. **Serveereinstellungen:** Manchmal kann es an den Servereinstellungen liegen. Überprüfen Sie, ob es irgendwelche Änderungen oder Wartungsarbeiten am Server gegeben hat, die die Funktionalität des Formulars beeinflussen könnten.
3. **E-Mail-Filter:** Auch wenn Sie bereits den Spam-Ordner überprüft haben, wäre es sinnvoll, die Einstellungen Ihres E-Mail-Anbieters auf mögliche Filter oder Blockierungen zu überprüfen, die die Zustellung

Save as Draft

Clear

DERSELBE USE CASE, DREI ITERATIONEN (ANONYMISIERT)

Prototyp 1 → 2 → 3: so wird er besser.



Nicht der erste Wurf zählt, sondern wie schnell Sie iterieren können.

Ein Prototyp, über den IT, Fachbereich und Geschäftsführung sprechen können.

Das Gegenmittel gegen monatelanges Spezifikations-Ping-Pong.

WICHTIGES PRAXIS-LEARNING

**KI fühlt sich sofort genial an.
Die Probleme zeigen sich erst in der Nutzung.**

Deshalb: Lieber einfache Use Cases zum Erfolg führen und jeden Prototyp mit **echten Nutzern** validieren.

Ergebnis Schritt 2

**Der Use Case ist validiert.
Jetzt darf Technik ins Spiel.**

3

SCHRITT 3 VON 4

Enterprise Readiness: Tooling & Setup

Guardrails statt Zeile-für-Zeile-Review.

KI-EINSATZ IST EIN SPEKTRUM

Drei Einsatzstufen.

Stufe 1: Prototyping

KI nur für Durchstiche und Konzeptvalidierung. Produktion bleibt klassisch. Schnellster, sicherster Start.

Stufe 2: Assistiert

KI schreibt mit Tests, Doku, Refactoring. Alles unter Review-Pflicht. Der Sweet Spot für die meisten Teams.

Stufe 3: KI-first

Die gesamte Pipeline KI-gestützt, Human-in-the-Loop an definierten Gates.

Welche Stufe passt, ist eine Reifegradfrage, keine Glaubensfrage.

KI-VOM VERSUCH ZUR PROFILÖSUNG

Vibe Coding. AI-Assisted. Agentic Engineering

Vibe Coding

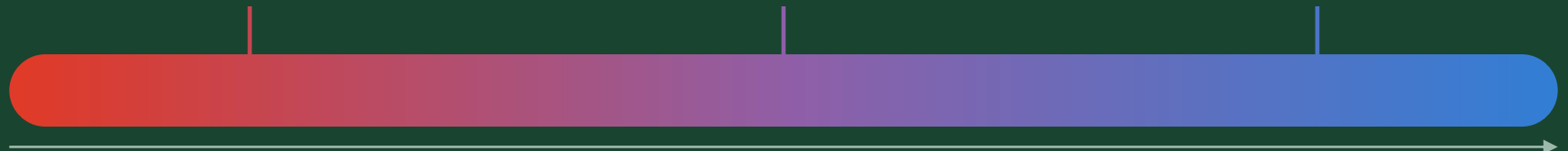
Lockere Prompts
«Scheint zu laufen?»
Wegwerf-Code

Strukturiert KI-gestützt

Prompts mit Vorgaben
Manuelle Tests, Stichproben
Features in bestehendem Code

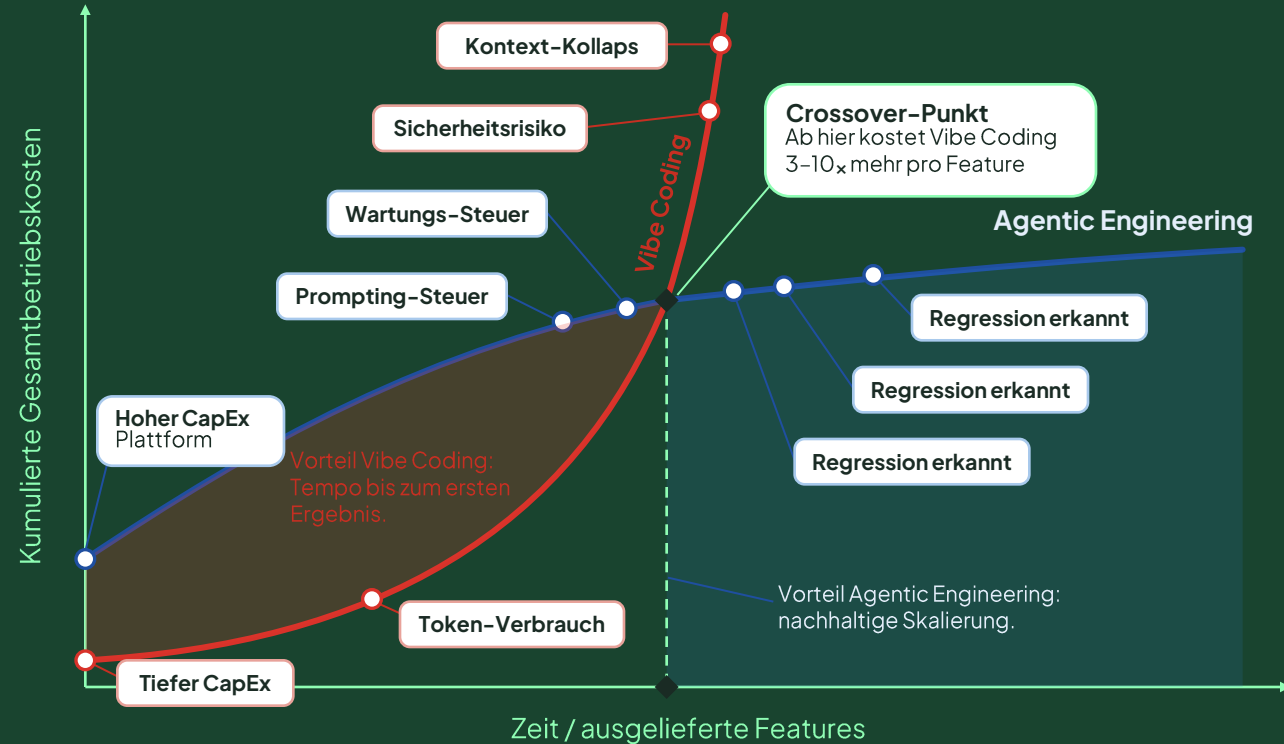
Agentic Engineering

Formale Specs & Architektur
Automatisierte Evals, CI/CD
Skalierende Produkktivsysteme



Weniger Struktur, mehr Tempo

Mehr Struktur, mehr Verlässlichkeit



Vibe Coding

12%

CapEx

Minimale Investition

OpEx

Hohe laufende Kosten

- Schnelles Prototyping, langsames Skalieren
- Hohe Reibung bei langfristiger Wartung
- Wirtschaftliche Sackgasse bei Komplexität

Agentic Engineering

12%

CapEx

Vorab-
Plattformdesign

OpEx

Tiefe marginale
Betriebskosten

- Kontrollierte Iteration, schnelles Skalieren
- Geringe Reibung bei automatischen Updates
- Nachhaltig bei reifen Codebasen

Entscheidung getroffen

**Sie gehen für
Agentic Engineering.**

DIE KOMBINATION, DIE SICH BEWÄHRT HAT

Das Referenz-Setup: Drei Bausteine. Ein System.

Ein Coding-Harness

Claude Code als Standard statt 5 Tools parallel. Ein Harness, den das ganze Team beherrscht.

Klare Repo-Struktur

Versionierung, Branch-Regeln, Review-Pflicht
KI-Code durchläuft dieselben Gates wie Menschen-Code.

Human-in-the-Loop

Definierte Freigabepunkte: Architektur, Datenmodell, Merge.
KI schlägt vor. Der Mensch entscheidet.

DIE LEITPLANKEN, DIE DAS TRAGEN

Guardrails statt Zeilen-Review.

Architektur-Tests

Prüfen automatisch, dass generierter Code die vorgegebene Architektur einhält – nicht, wie er formuliert ist.

Security-Tests

Secrets, bekannte Lücken und Abhängigkeiten: automatisiert bei jedem Pull Request.

Testabdeckung

Die KI beweist, dass die Funktionalität tut, was im Issue steht. Sie klicken durch, statt Code zu lesen.

Ehrlich: Einen Industriestandard gibt es dafür heute noch nicht.

KI-FIRST DEVELOPMENT KONKRET

So arbeitet ein KI-first Team.

- 1 KI schreibt zuerst: Jede Aufgabe geht als Issue an den Agenten
Pull Requests kommen zurück. Niemand startet mehr im leeren Editor.
- 2 Der Mensch führt: Architektur, Prioritäten, Freigaben
Entwickler werden zu Reviewern und Systemdesignern.
- 3 Guardrails sichern
Architektur-, Security- und Abdeckungstests prüfen jeden Vorschlag automatisch.

WIE MAN DAS ÜBERHAUPT FESTLEGT

Architektur, Guardrails und Messung.

Architektur

So festgelegt: als Regeldatei im Repo, die der Agent bei jedem Task liest.

Beispielregeln

- Business-Logik nur im Service-Layer
- Keine SQL-Queries im Controller
- Jede API-Route validiert ihren Input

Guardrails

So festgelegt: als automatische Checks, die einen Pull Request blockieren.

Beispielchecks

- Architektur-Test: Controller darf DB-Layer nicht importieren
- Security: keine Secrets im Code, keine kritischen Abhängigkeiten
- Testabdeckung mindestens 80 %

Messung

So festgelegt: Startwert vor der Einführung festhalten, dann dieselben Zahlen pro Sprint.

Beispielkennzahlen

- Zeit von Issue bis Merge
- Anteil KI-generierter Pull Requests
- Fehlerrate in Produktion vorher/nachher

Alles lebt im Repository: als Regeldatei, als Test und als Zahl. Nicht in einem Konzeptsdokument.

Guardrails as Code, versioniert im Projekt

CLAUDE.md

Regeln, Architektur

Agent-Settings

Erlaubte Befehle

CI-Workflows

Tests, Lint, Scans

Testsuite

Deckt Regressionen ab

Lint und Typecheck

Stil und Typen

Devcontainer

Sandbox des Agenten

Pull request

(Teil-)Review vor dem Merge

Branch protection

Kein direkter Push auf main

Required status checks

Auf Org-Ebene erzwungen

CODEOWNERS

Mensch gibt Pipeline und Tests frei

Secrets und Hosting

Vault, Modellwahl, Retention

Guardrails ausserhalb der Schreibrechte des Agenten

EIN KI-BILD MUSS SEIN

Bot hinter Mauern



SO SIEHT SO EINE REGELDATEI AUS

CLAUDE.md. Die Regeln, die der Agent immer liest.

Architektur

- Business-Logik ausschliesslich in /src/services
- Controller ohne SQL-Queries, nur Service-Aufrufe
- Jede API-Route validiert ihren Input über ein Schema

Definition of Done

- Unit-Test für jeden neuen Service
- Testabdeckung mindestens 80 %
- Keine neue Abhängigkeit ohne Rückfrage

Vorgehen

- Plan vor der Umsetzung im Issue kommentieren
- Ein Pull Request pro Issue, maximal 400 geänderte Zeilen

Eine Datei, im Repository versioniert. Jede Regel, die hier fehlt, muss ein Mensch in jedem Pull Request von Hand prüfen.

ZIEL, KONTEXT, AKZEPTANZKRITERIEN

Behandle die KI wie (unselbständigen) Entwickler

BEISPIEL 1 · ISSUE (BUG)

BUG-247: Bestellbestätigung zeigt falsche MwSt bei Rabattcodes

Beschreibung: Bei eingelöstem Rabattcode wird die MwSt auf den Originalpreis statt auf den reduzierten Preis berechnet. Standardbestellungen sind korrekt.

Reproduktion: Produkt für CHF 100 in den Warenkorb, Code SOMMER20 einlösen, Checkout abschliessen. Erwartet: 8.1 % MwSt auf CHF 80. Tatsächlich: MwSt auf CHF 100.

Akzeptanzkriterien

- MwSt wird auf den rabattierten Betrag berechnet
- Bestehende Unit Tests laufen grün, neuer Testfall für das Rabatt-Szenario ergänzt

BEISPIEL 2 · FEATURE REQUEST

FEAT-312: Bestellstatus per WhatsApp abfragen

Beschreibung: Kunden fragen ihren Bestellstatus per WhatsApp ab, statt im Kundencenter anzurufen. Ziel: geschätzt 30 Anrufe pro Woche weniger im Support.

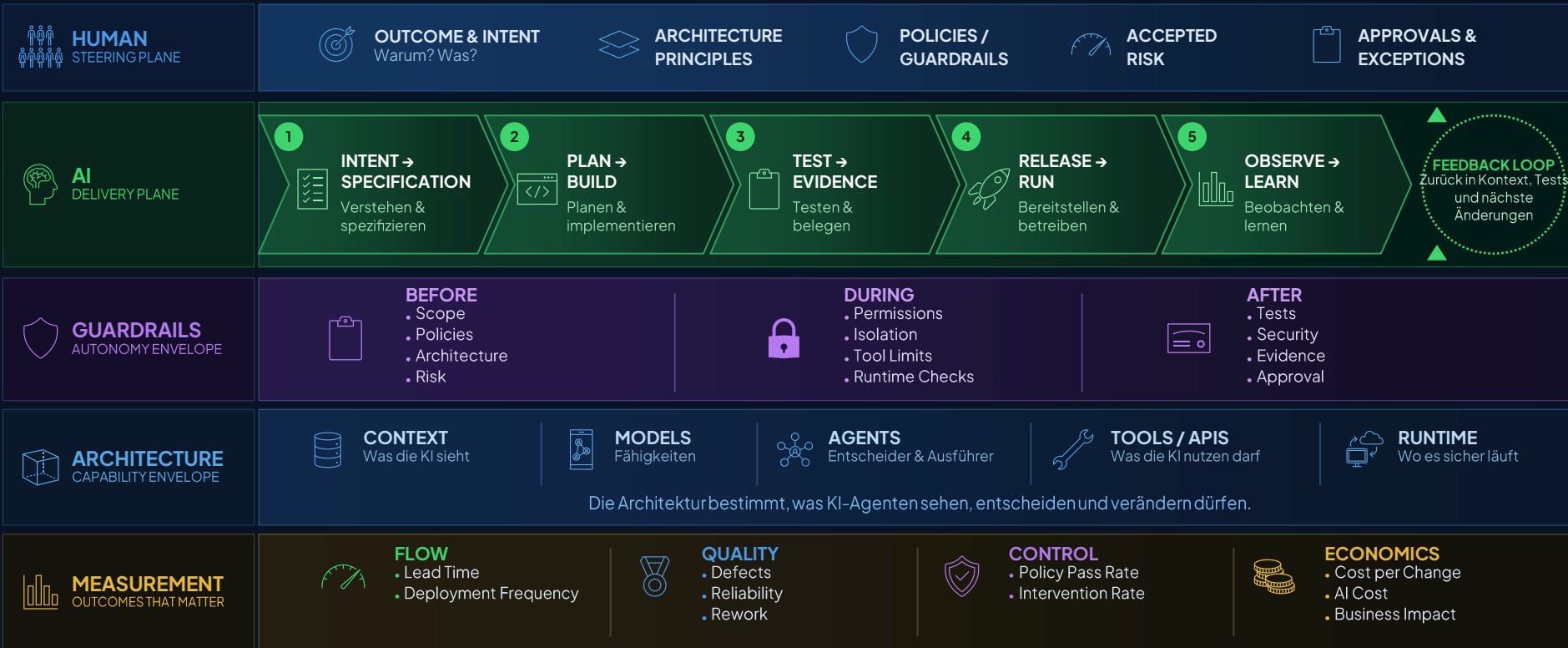
Umsetzung: Neuer Endpoint /api/orders/status, Anbindung an die bestehende WhatsApp Business API. Kunde sendet die Bestellnummer, erhält Status und voraussichtliches Lieferdatum.

Akzeptanzkriterien

- Abfrage funktioniert mit gültiger Bestellnummer
- Ungültige Nummern erhalten eine freundliche Fehlermeldung
- Keine Herausgabe von Kundendaten ohne Verifikation (PLZ-Abgleich)

AI-First Development – Von der Idee bis zur Produktion

Ergebnisorientiert. Mit klaren Leitplanken. Messbar.



“

In AI-first Development ist Code nicht mehr das primäre Arbeitsergebnis.
Das Arbeitsergebnis ist ein **kontrollierter Strom von Intent zu produktiver Veränderung** – inklusive Evidenz.

”

Von einfacher KI-Entwicklung zu echter Enterprise AI Readiness

Der Unterschied: einzelne Produktivität vs. ein KI-fähiges Entwicklungssystem

Einfache KI-Entwicklung

KI Unterstützt einzelne Entwickler

-  Coding-Hilfe
-  Einzelne Prompts
-  Wissen im Kopf
-  Repository-Zugriff
-  Review vor allem durch Menschen
-  Einzelne Tools
-  Lokale Produktivität
-  Erfolg = schneller programmieren
-  Optimierte einzelne Entwickler

Reifegrad

Enterprise AI Readiness

KI liefert kontrolliert entlang der Delivery Pipeline

-  End-to-End Delivery
-  Intent → Spec → Build → Test → Release → Observe
-  Kontext systematisch verfügbar
-  Kontrollierter Zugriff auf Tools, APIs, CI/CD & Cloud
-  Policies + Evidence + Security
-  Orchestrierte Agents + Runtime
-  Messbare Wirkung
-  Erfolg = schneller und sicherer produktive Änderungen liefern
-  Verändert das gesamte Entwicklungssystem

5 Readiness-Dimensionen



1. Context Readiness

Versteht die KI die Domäne, Architektur und Entscheidungen?



2. Agent & Tool Readiness

Kann die KI handeln, testen und integrieren?



3. Governance Readiness

Ist klar, was die KI darf?



4. Delivery Readiness

Kann Änderung von Intent bis Production laufen?



5. Measurement Readiness

Wird Flow, Quality, Control und Economics gemessen?



Copilot Readiness = KI kann Code schreiben.

Enterprise AI Readiness = KI kann verantwortbar Veränderungen liefern.

Ergebnis Schritt 3

**Die Leitplanken stehen.
Jetzt muss es beim Team ankommen.**

4

SCHRITT 4 VON 4

Team Enablement

Der Funke muss überspringen. Kultur und Change entscheiden, nicht das Tool.

Pilot-Gruppe statt Big Bang.

Allen alles freischalten und hoffen,
das ist keine Einführung, das ist Chaos mit Ansage.

DAS PRINZIP, DAS ECHTE ANWENDUNG AUSLÖST

Nicht schulen. Gemeinsam bauen.

Setzen Sie sich mit 1–2 Key-Entwicklern hin und baut den ersten Use Case zusammen. Repo aufsetzen, Prinzipien formulieren, testen. Dann macht es Klick.

DIE STRUKTUR DAHINTER

Die KI-Dev-Pilotgruppe.

Wer?

1–2 KI-affine Entwickler als «Dev Champions». Sichtbar benannt, mit Zeit ausgestattet, wertgeschätzt.

Was?

Sie bauen die ersten Use Cases, dokumentieren die Learnings und definieren die KI-Prinzipien fürs ganze Team.

Warum wirkt es?

Die anderen sehen echte Ergebnisse von echten Kollegen im eigenen Codebase und sagen: «Will ich auch.»

KI-Einführung braucht Multiplikatoren (Evangelisten), keine Verordnung von oben.

SO KOMMT KI INS TEAM

Der Rollout-Flow.

2-3 Key-Player
statt Big Bang

+

Gemeinsam bauen
statt schulen

+

Erfolge
sichtbar machen

+

C-Level
Attention & Decision

=

Team für Team ausrollen
mit Erfahrungsaustausch und interner Community.

DIE FRAGE, DIE IHR TEAM WIRKLICH STELLT

«Ersetzt das uns als Entwickler?»

Was sich verschiebt

Wer bisher vor allem Boilerplate getippt hat, verliert einen Teil seiner Arbeit. Das ist real. Es hilft niemandem, das im Team wegzureden.

Was mehr wert wird

Architektur, Domänenwissen, ein präzise formuliertes Issue, Urteilsvermögen im Review. Genau die Fähigkeiten, für die vorher nie Zeit war.

Achtung: Juniors

Wer mit Agenten startet, bevor er den Code versteht, lernt es nie. Legen Sie fest, was Junioren ohne Agent bauen. Sonst fehlen Ihnen in fünf Jahren die Seniors.

Diese Antwort entscheidet über Schritt 4: ein Team mit Angst wendet das Werkzeug nicht an, es umgeht es.

Das Ziel: Love for AI-first Development

Dann, und erst dann, ist KI in der Entwicklung produktiv.
Und die Schatten-Nutzung erledigt sich von selbst.

Ergebnis Schritt 4

KI wird aktiv eingesetzt.

Nun muss es zum Alltag werden.

DER KOMPLETTE FAHRPLAN

Vier Schritte. Eine klare Reihenfolge.



Prozesse verstehen: Potenziale mappen, dann den Detailprozess erfassen



Prototyping: vertikal, mit Echtdaten – Claude Code + GitHub



Enterprise Readiness: Guardrails, automatisierte Tests



Team Enablement: Pilotgruppe, Erfolge sichtbar, C-Level-Entscheid

WO KI HEUTE SCHON SEHR STARK IST

Erweiterungen in bestehenden Systemen.

«Ich hätte gerne eine eingabemaske um schneller und in einem einzigen schritt in microsoft dynamics firmen und kontakte inkl. lead oder opportunity anzulegen.»

Unser internes Customer Center ist zugleich Spielwiese für unsere KI-Entwicklung: Lead und Kontakt gleichzeitig erfassen. Ein einfacher Prompt – die KI wusste, wo der Button hingehört, und hat die Funktion selbst entwickelt.

Je klarer das bestehende System, desto treffsicherer die KI-Erweiterung.

WO KI HEUTE SCHON SEHR STARK IST

Erweiterungen in bestehenden Systemen.

10 CUSTOMER CENTER

Navigation

- Dashboard
- Customers
- Projekte
- Solutions
- Tenders
- Sales
- Angebote
- Net Promoter Score
- 360° Feedback

Ask OPTony

 Hali Galli

Customers

Search customers...

Filter Active Projects

Neu anlegen

Company Name	Contact	Adress	Responsible	Active Projects
Adatum Corporation	—	Redmond	Severin d'Heureuse	0 >
Adventure Works Cycles	—	Santa Cruz	Severin d'Heureuse	0 >
Alpine Ski House	—	Richmond	Severin d'Heureuse	0 >
Bel lows College	—	Chicago	Severin d'Heureuse	0 >
Best For You Organics Company	—	Pittsburgh	Severin d'Heureuse	0 >
Contoso, Ltd.	—	Los Angeles	Severin d'Heureuse	0 >
Contoso Suites	—	Seattle	Severin d'Heureuse	0 >
Consolidated Messenger	—	Richmond	Severin d'Heureuse	0 >
Fabrikam, Inc.	—	San Francisoc	Severin d'Heureuse	0 >
Fabrikam Residences	—	Lynnwood	Severin d'Heureuse	0 >
First Up Consultants	—	New Jersey	Severin d'Heureuse	0 >
Fourth Coffee	—	Redmond	Severin d'Heureuse	0 >

1-100 of 945

2
3

DIE EHRliche SLIDE. WO KI-ENTWICKLUNG HEUTE SCHEITERT.

Schnell generiert ist nicht sauber gebaut.

Frontend-Qualität

UX/UI-Fehler, Overflows, kaputte Umbrüche
Frontend ist noch nicht die Stärke der KI

Dead Code & fehlende Doku

Nach 3 Iterationen weiss niemand mehr, was wofür da ist

Ownership- & IP-Verlust

Das Team versteht den eigenen Code nicht mehr
Das Wissen liegt beim Prompt-Verlauf

3× Aufwand bei Iteration

Erster Wurf in Stunden
Aber unstrukturierte Weiterentwicklung kostet dreifach

WO WIR KRACHEND GESCHEITERT SIND.

Erfahrungen aus dem Alltag.

Verlust von 2.5 Monaten harter Arbeit

Infrastructure by Code löschte alle Medienfiles unwiderruflich.

2 x so viel Aufwand bei Standardprozess

KI-generierte Website war First-Shot super. Erweiterungen waren nicht möglich. KI auf Abwegen.

Abstürze nach Going Live

In einer Produktionsstörung lieferte die KI mehrfach plausibel klingende, aber völlig falsche Diagnosen.

DER BLICK NACH VORN

Wenn niemand mehr jede Zeile lesen kann.

Heute: Der Mensch prüft

Code wird über Pull Requests geprüft und freigegeben. Das funktioniert. Noch.

Morgen: KI prüft KI

Generierter Code wird von KI verifiziert, nach definierten Massstäben und Vorgaben.

Offen: Forschung uneins

Der aktuelle Stand der Forschung ist sich uneins. Niemand weiss heute, in welche Richtung es geht.

Darum bauen wir die Guardrails schon heute auf damit Vertrauen nicht an der einzelnen Codezeile hängt.

Code, den kein Mensch mehr im Detail versteht? Eine sehr valable Zukunftsaussicht.

Niemand kann heute seriös beurteilen, in welche Richtung es geht. Wir auch nicht.

Aber wer Prozesse, Tests und Guardrails heute sauber aufsetzt, ist deutlich besser vorbereitet.

**Lassen Sie uns das einmal kurz
zusammenfassen.**

DER VALUE-STACK

Heute zeigten wir Ihnen ...



... wie Sie identifizieren, wo KI in Ihrer Entwicklung wirkt.



... wie Sie mit vertikalen Prototypen Use Cases in Stunden validieren.



... wie Sie das Ganze Enterprise Ready machen: Guardrails, automatisierte Tests.



... wie KI wirklich im Team ankommt. Pilotgruppe, gemeinsam bauen, Prinzipien. Spass!

DER VALUE-STACK

Heute zeigten wir Ihnen NICHT ...



... wie Sie Kosten durch unterschiedliche Modelle sparen können.



... wie Sie Ihre Effizienzsteigerung messen.



... wie Sie vermeiden mit KI in Sackgassen zu geraten.



... wie Sie die Guardrails konkret aufsetzen.

**Würden Sie sagen, unsere gemeinsame
Zeit heute war wertvoll?**

**Sehen Sie, wie ein oder mehrere Punkte
auch bei Ihnen in der Praxis funktionieren
könnten?**

Denken Sie aber, dass es super viel Aufwand ist die vier Schritte umzusetzen, das Setup aufzubauen und die Dev Champions zu enablen?

Dann brauchen Sie uns! :-)

WAS OPTEN IST. IN EINEM SATZ.

OPTEN ist die KI-first Tech Unit, die alles umsetzt, was Ihre interne IT alleine nicht schafft. Vom Prototyp bis zur Plattform.

Nicht gegen Ihre IT. Mit Ihrer IT. Seit 17 Jahren, Swiss Made.

GENAU DIESE PHASEN – GEMEINSAM MIT UNS

So unterstützt OPTEN in jeder Phase.

1 · Prozesse verstehen

Discovery- und Prozess-Workshops: Pain Point Mapping, Detailprozess, Business Case.

2 · Prototyping

Vertikale Prototypen mit Echtzeiten in Tagen – inklusive Setup mit Claude Code und GitHub.

3 · Enterprise Readiness

Guardrails, automatisierte Tests und Integration in SAP, ABACUS, Dynamics, APIs.

4 · Team Enablement

Webcoding-Workshops, Pilotgruppen, KI-Prinzipien und Change-Begleitung für Ihre Entwickler.

Einzel buchbar oder als kompletter Weg von der Idee bis zur Plattform.

UND ALLES DAVON: SOUVERÄN

Sicherheit ist eingebaut, nicht angeflanscht.

Swiss Made Software

Entwickelt in Zürich-Stettbach.
DSGVO-konform. Ihre Daten
bleiben, wo sie hingehören.

Ihre IP. Ihr Code.

Dokumentiert, versioniert,
übergeben. Kein Vendor-Lock-in,
Ihr Team versteht, was gebaut
wurde.

Bewährte Partner

Microsoft-Stack, Azure, Umbraco
SilverPartner. Enterprise-
Standards statt Experimente.

Möchten Sie mehr darüber erfahren?

DER NÄCHSTE SCHRITT

Ihr kostenfreies KI-Strategiegespräch.

20 Min, online & kostenlos

Standortbestimmung:
Wo steht Ihre
Entwicklung?
Schatten-Nutzung,
erste Tools, laufende
Projekte?

2–3 Use-Case- Empfehlungen

Konkret für Ihr Team,
mit ROI-Berechnung.

Ehrliche Empfehlung

Passt OPTEN? Wenn
nicht, sagen wir das.
Kein Pitch, eine
Arbeitssitzung. 5 Plätze
dieses Quartal.

Anmelde- Link



www.opten.ch/digi-check *Link kommt parallel in den Chat.*

JETZT SIND SIE DRAN

Q&A. Ihre Fragen.

Häufige Fragen

Welche weiteren Tools empfiehlt
ihr konkret? · Wie schätzt ihr die
chinesischen Modelle ein? · Was
kostet ein Prototyp?

... und weiter

Wie schätzt ihr die Zukunft
allgemein ein? · Wie überzeugen
wir die Geschäftsleitung? · Was ist
der nächste grosse Schritt?

Termin-Link

Digi Check:
www.opten.ch/digi-check
Noch 5 Plätze für dieses Quartal.

VIELEN DANK · LIVE-WEBINAR · 19.08.2026

KI in der Softwareentwicklung.

Wie IT-Teams jetzt wirklich entlastet werden:
Der Fahrplan in 4 Schritten

Tobias Morf (Gründer & Inhaber, OPTEN AG) · www.opten.ch/digi-check