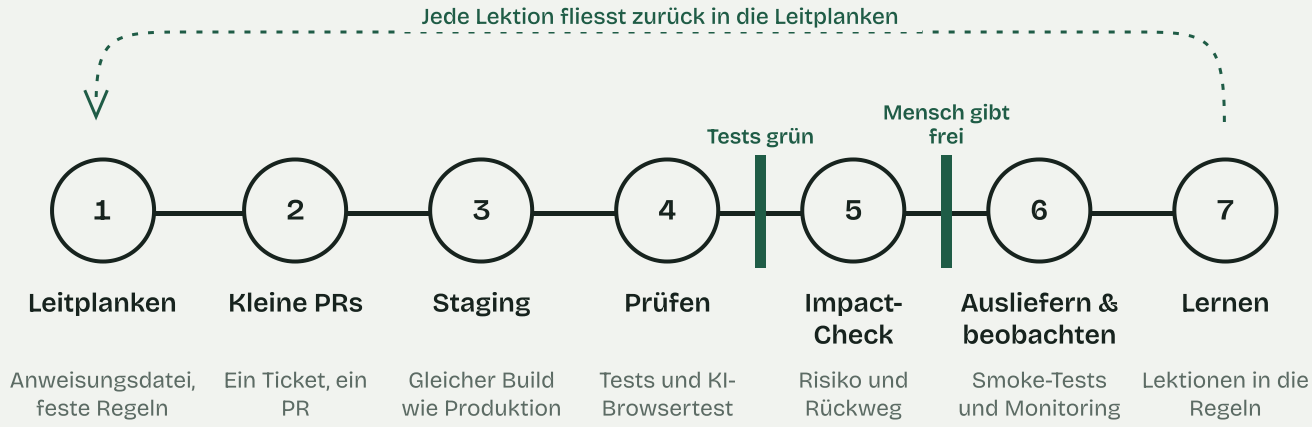


Dominik Heinrich, Multi-Cloud DevOps & AI Solution Architect

Mit KI-Coding-Agenten sicher ausliefern

Ein Einführungs-Framework für einen KI-gestützten Software-Delivery-Lifecycle (AI-SDLC). Es ist der Prozess hinter über 5'000 gemergten Pull Requests auf zwei produktiven Plattformen.



Sieben Stufen, zwei harte Gates. Agenten schreiben den Code, Menschen geben frei, was in Produktion geht.

Ein produktives SaaS, gebaut mit diesem Prozess, anonymisiert

3'025	1'048	160	3
gemergte Pull Requests in zehn Monaten	automatisierte Testdateien	Serverless Functions	Apps: Web, iOS und Android

Die sieben Stufen

Einmal pro Team und Repository einrichten. Abhaken, sobald sie live sind.

01 Zuerst die Leitplanken schreiben

- [ ] Eine kurze Anweisungsdatei pro Repository: Architektur, Konventionen, was der Agent darf und was nicht.
- [ ] **Feste Regeln:** Login, Zahlungen, Datenlöschung und Secrets brauchen immer eine menschliche Freigabe.
- [ ] Agenten arbeiten nie mit Produktivdaten oder echten Kundenkonten, nur mit Testkonten.

02 Jede Änderung klein und umkehrbar halten

- [ ] Ein Ticket, ein Pull Request. Branch Protection und Arbeit nur über PRs, keine direkten Pushes.
- [ ] Datenbank-Migrationen idempotent und vor dem Merge im Trockenlauf gegen die Zieldatenbank geprüft.

03 Immer zuerst Staging

- [ ] Jede Änderung läuft vor der Produktion über eine Staging-Umgebung, mit demselben Build und derselben Konfiguration.
- [ ] Nur geänderte Functions oder Services werden deployt, damit jedes Release klein bleibt.

04 Prüfen wie ein Nutzer

- [ ] Unit- und End-to-End-Tests, dazu Typecheck- und Lint-Baselines, die nur besser werden können.
- [ ] Ein KI-Browsertest in echtem Chrome: der Ablauf in Alltagssprache beschrieben, Ergebnis mit Fehlern und Fix-Prompt.

05 Die Auswirkung vor der Produktion analysieren

- [ ] **Eine kurze Tabelle pro Release:** was sich ändert, wer betroffen ist, das Risiko und der Rückweg.
- [ ] Ein Mensch gibt den Merge in die Produktion frei.

06 Ausliefern und beobachten

- [ ] Smoke-Tests auf kritischen Routen nach jedem Deploy, und der Deploy wird rot, wenn eine ausfällt.
- [ ] Ergebnis-Monitoring und pro Fehler ein automatisch eröffnetes Issue, das sich bei Erholung wieder schliesst.

07 Den Kreis schliessen

- [ ] Kundenantworten entwirft der Agent, abgeschickt werden sie von einem Menschen.
- [ ] Jede Erkenntnis fliesst zurück in die Anweisungsdateien, so verbessert sich der Prozess selbst.

Einführung nach Unternehmensgrösse

Startup: Leitplanken, Staging und Browser-Checks in ein bis zwei Wochen. Tempo kommt aus dem Ablauf, nicht aus dem Überspringen.

KMU: ein Pilotteam für vier bis sechs Wochen, ein Security-Review, dann die Einführung in den anderen Teams.

Konzern: zuerst Governance: eine freigegebene Liste von Modellen und Anbietern, Datenklassifizierung, lokale Modelle für sensiblen Code, ein Audit-Trail und ein Plattform-Team, das die anderen befähigt.

Was man messen sollte

Durchlaufzeit vom Ticket bis zur Produktion, Change-Failure-Rate, gemergte Pull Requests pro Woche und von Nutzern gefundene Fehler.

Die häufigsten Fehler

- Agenten pushen direkt auf den Main-Branch oder haben Zugangsdaten für die Produktion.
- Pull Requests mit Tausenden Zeilen, die niemand wirklich prüfen kann.
- Staging überspringen, weil die Tests grün sind.
- Keine Anweisungsdatei, also beginnt jede Session bei null und wiederholt alte Fehler.
- Codezeilen messen statt Durchlaufzeit und Change-Failure-Rate.

So arbeiten wir zusammen

Dieselbe Qualität für ein Startup mit zehn Leuten und einen globalen Konzern. Das Format passt sich Ihnen an.

Startups

Fractional DevOps- & KI-Lead

Ein paar Tage im Monat: Architektur, Release-Pipeline und KI-Funktionen, damit sich die Gründer auf Kunden konzentrieren können. Vom Prototyp in die Produktion, ohne Neubau.

KMU und Mittelstand

Ein Projekt mit klarem Ergebnis

Ein klares Ziel wie tiefere Cloud-Kosten, eine Migration oder die ersten KI-Agenten in Produktion, mit festem Umfang, wöchentlichem Fortschritt und einer Übergabe, die Ihr Team betreiben kann.

Konzerne

Ein Architekt in Ihren Teams

Eingebettet in Plattform-, Security- und Produktteams: Architekturentscheidungen, Governance für KI und regulierte Umgebungen. Für grössere Vorhaben bringe ich ein erfahrenes Senior-Team mit.

Wie ich arbeite

Entscheidungen schriftlich: Architecture Decision Records mit den Abwägungen, damit Ihr Team weiss, warum, nicht nur was.

Sicher ab Werk: OIDC und Managed Identities, Least Privilege, Secrets im Vault, ein Audit-Trail.

Betrieben, nicht nur geliefert: SLOs, Alerts auf Geschäftsergebnisse und eine Incident-Routine.

Gemessen: Kosten, Durchlaufzeit, Verfügbarkeit und gesparte Stunden, jede Woche berichtet.

Das Wissen bleibt bei Ihnen: Runbooks, Dokumentation und Workshops für Ihr Team.

Ein Ansprechpartner, der mit Engineering und Management spricht, auf Deutsch, Englisch und Spanisch.



Soll das in Ihrem Team laufen?

30-Minuten-Call buchen

E-mail mail@dominikheinrich.ch
Web dominikheinrich.ch

Weltweit verfügbar, remote. Vor Ort auf Anfrage, je nach Land.