

## 01 / EINRICHTEN

# Baue die Funktion, für die du **zahlst.**

Replica verteilt den App-Nachbau auf elf Claude-Skills. Starte mit einer Kernfunktion: zum Beispiel einer eigenen Seite für Erstgespräche.



CLAUDE CODE + REPLICA · DEUTSCHE FASSUNG 1.0.1

## 01 Claude Code öffnen.

Installiere Claude Code nach der offiziellen Anleitung, melde dich an und öffne einen neuen, leeren Projektordner, z. B. **meine-terminseite**.

Voraussetzung: Git im System. Für die Demo: Python 3.8+ oder ein vorhandener lokaler Webserver. Die Hilfsskripte brauchen Python.

## 02 Deutsche Fassung installieren.

Gib die Befehle einzeln in Claude Code ein:

```
/plugin marketplace add neubauerwilli/replica-skill-deutsch
```

```
/plugin install replica-skill-deutsch@replica-skill-deutsch
```

Im Installationsdialog den Geltungsbereich wählen, installieren und unter **/plugin** → **Installed** den Status prüfen.

## 03 Mit einem kleinen Auftrag starten.

Öffne die Terminseiten-Vorlage und kopiere den vollständigen Startprompt. Der Fall auf Seite 3 zeigt, was daraus entstehen soll.

Das Skill-Paket ist kostenlos. Claude-Zugang, Hosting und weitere Dienste haben eigene Konditionen. Ein Nachbau braucht Prüfung; identische Funktionen sind nicht garantiert.

[Deutsche Fassung ↗](#)   [Original von Jake ↗](#)

# Elf Skills. Ein gemeinsamer Bauplan.

Du wählst den nächsten Schritt. Nicht jeder Auftrag braucht sofort Backend, Marke und Livegang.

VERSTEHEN & PLANEN

|           |                                                                             |
|-----------|-----------------------------------------------------------------------------|
| Erkundung | Untersucht zugängliche Referenzseiten und hält Funktionen und Abläufe fest. |
| Architekt | Legt Aufbau, Datenmodell und technische Entscheidungen fest.                |
| Design    | Erfasst Layout, Typografie, Farben und wiederverwendbare Bauteile.          |

UMSETZEN & PRÜFEN

|           |                                                                             |
|-----------|-----------------------------------------------------------------------------|
| Bau       | Setzt die Oberfläche Bildschirm für Bildschirm um.                          |
| Backend   | Ergänzt echte Daten, Authentifizierung und Serverlogik, wenn nötig.         |
| Test      | Prüft Bedienung, Fehlerfälle und Abläufe im Browser.                        |
| Vergleich | Vergleicht Referenz und eigene Oberfläche und sucht sichtbare Abweichungen. |

WEITERENTWICKELN

|             |                                                                          |
|-------------|--------------------------------------------------------------------------|
| Unternehmer | Recherchiert Markt und Nutzerprobleme; leitet belegte Verbesserungen ab. |
| Marke       | Entwickelt eine eigene Identität für deine Version.                      |
| Start       | Bereitet den Marktstart und die Kommunikation vor.                       |
| Livegang    | Bereitet Hosting, Veröffentlichung und Betrieb vor.                      |

Im Slash-Menü stehen die Namen mit **replica-** davor. Der volle Aufruf für den Einstieg lautet:

/replica-skill-deutsch:replica-erkundung

Alle elf kopierbaren Aufrufe findest du auf der [Begleitseite](#).

# Aus einer Buchungsfunktion wird deine **Terminseite**.

Fiktives Beispiel: „Nordlicht Beratung“ möchte 30-Minuten-Erstgespräche lokal im Browser buchen lassen.

Nordlicht Beratung

DEMO

Dein Erstgespräch · 30 Minuten

Dienstag, 27.10.2026 · Europe/Vienna

09:00

10:00 ✓

11:00

Mia Muster

mia@example.com

Termin bestätigen

Erwartet: Bestätigung, lokal gespeicherte Buchung und dieser Slot nach Neuladen weiterhin gesperrt.

ERKUNDEN

→ PLANEN

→ BAUEN

→ TESTEN

## Gib Material und Grenzen mit.

Nutze die öffentliche [Cal.com-Seite](#) als Funktionsreferenz oder beschreibe deinen eigenen Buchungsablauf. Die Demo erhält eigenes Design und nur die drei festen Termine.

STARTAUSZUG / VOLLSTÄNDIGEN PROMPT ONLINE KOPIEREN

Du unterstützt mich als App-Entwickler. Ich arbeite mit Claude Code in einem leeren Projektordner. Baue für die fiktive „Nordlicht Beratung“ eine lokale Terminseite mit drei Slots am 27.10.2026 um 09:00, 10:00 und 11:00 Uhr, jeweils 30 Minuten, Europe/Vienna. Name und E-Mail sind Pflicht. Speichere lokal im Browser. Keine echten Mails, Zahlungen oder Fremd-Logos.

Öffne die [Vorlage mit Start- und Korrekturprompt](#) ↗. Sie verlangt Browserprüfungen und höchstens zwei Korrekturrunden. Diese Seite ist eine erklärte Musteransicht; die Demo wurde in diesem Auftrag nicht von Replica fertig gebaut.

# Ein schöner Bildschirm ist erst der Anfang.

Lass den Test-Skill konkrete Handlungen ausführen. Verlange zu jedem Punkt: bestanden, fehlgeschlagen oder offen – mit beobachtetem Ergebnis.

- 01 **Buchen:** 10:00 wählen, Testname und example.com-Adresse eingeben. Stimmt die Bestätigung mit Datum und Zeitzone?
- 02 **Fehler provozieren:** Namen leer lassen und eine ungültige E-Mail eingeben. Keine Buchung; verständliche Meldung am Feld.
- 03 **Doppelt klicken & neu laden:** Nur ein gespeicherter Datensatz; 10:00 bleibt im selben Browser gesperrt.
- 04 **Handy & Tastatur:** Bei 375 px ohne horizontalen Bildlauf buchen; alle Felder und Buttons mit Tab erreichen.
- 05 **Grenze prüfen:** Ein zweiter Browser teilt diesen Speicher nicht. Echter Mehrnutzer-Doppelbuchungsschutz bleibt bis zum Backend offen.

## Der Unternehmer sucht den besseren Weg.

Lass ihn öffentlich belegte Beschwerden mit Quelle, Datum und Kontext auswerten. Keine erfundenen Rezensionen. Wähle eine Verbesserung und prüfe sie wieder im Browser.

### VORHER / FIKTIVER PROBLEMFALL

„10:00 buchen“  
Die Zeitzone fehlt. Der Termin bleibt für auswärtige Kunden unklar.

### NACHHER / PRÜFBARE ÄNDERUNG

„10:00 · Europe/Vienna“  
Datum und Zeitzone auch in der Bestätigung sichtbar.

Für einen öffentlichen Dienst zusätzlich Backend, Datenschutz, Rechte an Inhalten und Betrieb prüfen. Lokale Demo und Marktrecherche belegen noch keine verkaufsfertige App.

### Zur Community-Warteliste ↗

Freiwillig vormerken. Anleitung und Download bleiben frei zugänglich.