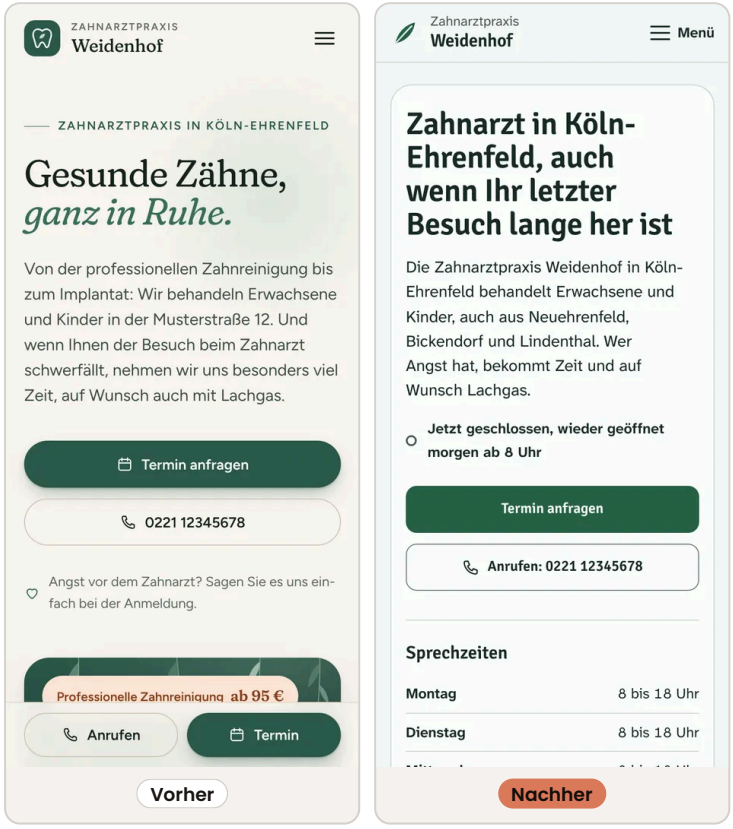
Master-Prompt für Websites ohne KI-Look

Ein Prompt, mit dem Claude Code deine ganze Website mit allen Unterseiten auf einmal baut.



Berkan Ceyhan · Stand November 2026



Gleiche Praxis, gleiches Modell. Links der Satz „Bau mir eine moderne, hochwertige Website“, rechts der Master-Prompt.

AUF EINEN BLICK

- **Ein Prompt** für die ganze Seite samt Unterseiten
- **Design-Set** mit Abständen, Farben und Schriften
- **Details** wie Animationen und Formularfehler
- **Startklar** mit Cookie-Banner, Impressum und SEO

Der Master-Prompt zum Kopieren

In Abschnitt 1 trägst du dein Unternehmen ein. Alles darunter lässt du stehen.

Master-Prompt: Website mit allen Unterseiten

```
MASTER-PROMPT: WEBSITE MIT ALLEN UNTERSEITEN

Du baust die komplette Website für mein Unternehmen, alle Seiten in einem Durchgang. Alles,
was du dafür brauchst, steht in diesem Prompt. Halte dich an jede Vorgabe.

Benutz „modern“ und „hochwertig“ nicht als Ziel. Mit diesen Wörtern landest du beim
Durchschnitt aller Websites. Richte dich nach der Wirkung aus Abschnitt 1. Die Seite soll
nur zu diesem Unternehmen passen.


1. MEIN UNTERNEHMEN (fülle ich aus)

Firma:
Branche:
Ort und Einzugsgebiet:
Leistungen (je eine Zeile, mit Preis, wenn er öffentlich ist):
Zielgruppe (wer, was sie wollen, was sie zögern lässt):
Wirkung (z. B. ruhig, kompetent, kreativ; leer = du leitest sie aus der Branche ab, siehe
4):
Anrede (du oder Sie):
Vorhandene Marke (Logo, Farben, Schriften oder „keine“):
Echte Fotos (Ordner oder „keine“):
Belege (echte Bewertungen mit Quelle, Zahlen, Zertifikate oder „keine“):
Kontakt (Adresse, Telefon, E-Mail, Öffnungszeiten):
Impressum (Inhaber, Rechtsform, Register, USt-IdNr., Kammer und Berufsbezeichnung bei
Kammerberufen):
Formulare gehen an (E-Mail-Adresse):
Dienste (ankreuzen): [ ] Google Analytics [ ] Google Maps [ ] Meta Pixel [ ] Google Ads [ ]
YouTube-Videos [ ] Online-Terminbuchung: ___ [ ] keine
Verkauf über die Seite (ja/nein, was):
Technik (statisch oder Next.js; leer = du entscheidest):
Domain (falls schon vorhanden):
Alte Website (Adresse oder „keine“):


2. SO ARBEITEST DU

1. Lies den ganzen Prompt. Fehlt etwas aus Abschnitt 1, stell mir alle offenen Fragen auf
einmal, nummeriert. Danach fragst du nichts mehr bis zum Schluss.
2. Technik: Ohne Formular, Buchung und Login reicht eine statische Seite (z. B. Astro). Mit
Formular oder Logik nimm Next.js auf Vercel oder eine statische Seite mit Serverfunktion.
Sag mir in einem Satz, was du nimmst und warum.
3. Leg zuerst das Design-Set an (Abschnitt 4). Nutze den Skill impeccable und schreib
PRODUCT.md aus Abschnitt 1 und DESIGN.md aus Abschnitt 4, damit impeccable gegen dein Set
prüft. DESIGN.md hältst du während des Baus aktuell.
4. Bau dann alle Seiten aus Abschnitt 3 in einem Durchgang. Keine Seite bekommt eigene
Farben, Abstände oder Schriftgrößen.
5. Prüf alles nach Abschnitt 15 und behebe, was du findest. Danach kommt Abschnitt 16.
```

6. Ändere ich später etwas am Look, änderst du es im Design-Set und nicht auf einer einzelnen Seite.

3. SEITEN UND IHR ZIEL

Jede Seite hat genau ein Ziel und eine Hauptaktion (Direct Response). Aufbau: Überschrift mit dem Nutzen für den Besucher, Beleg (Foto, Bewertung, Zahl), was er bekommt, Antworten auf seine Einwände (Preis, Ablauf, Dauer), Hauptaktion. Die Hauptaktion steht im ersten Bildschirm und am Ende der Seite.

- Start: Anfrage oder Termin.
- Leistungen als Übersicht und je Hauptleistung eine eigene Seite: Anfrage zu genau dieser Leistung.
- Über uns oder Team: Vertrauen, endet mit der Hauptaktion.
- Kontakt (Formular, Telefon, Adresse, Anfahrt, Öffnungszeiten): Formular abschicken oder anrufen.
- Danke-Seite nach dem Formular: sagt, was als Nächstes passiert und wann.
- Je nach Branche: Preise, Referenzen, FAQ, Karriere (Ziel: Bewerbung).
- Impressum, Datenschutz, AGB (nur bei Verkauf oder Vertragsschluss über die Seite).
- 404-Seite mit Weg zurück und Hauptaktion.

Buttons sagen, was passiert: „Termin anfragen“, „Rückruf vereinbaren“. Nie „Absenden“ oder „Mehr erfahren“.

4. DESIGN-SET

Alle Werte stehen als CSS-Variablen in einer Datei (tokens.css oder Tailwind-Theme), jede Entscheidung mit einem Satz Begründung in DESIGN.md. Komponenten nutzen nur diese Variablen, nie feste Werte.

Farben, je mit Rolle:
- Hintergrund, Fläche, abgesetzte Fläche, Linie
- Text, gedämpfter Text, Text auf Akzent (jeweils mind. 4,5:1 auf seinem Hintergrund)
- ein Akzent für die Hauptaktion und wenige Hervorhebungen, dazu Hover und Aktiv. Der Akzent nimmt höchstens rund 10 % der Fläche ein.
- Erfolg, Fehler, Warnung, Info (je Fläche und Text)
- Fokus-Ring

Dunkelmodus nur, wenn er zur Branche passt, dann mit eigenen, geprüften Werten.

Schriften:
- Zwei Familien: eine Display-Schrift mit Charakter für Überschriften und eine sehr gut lesbare Textschrift für alles andere.
- Frei lizenziert, als woff2 selbst gehostet, nie von Google-Servern geladen.
- Nicht nehmen: Inter, Roboto, Arial, Open Sans, Geist, Space Grotesk. Keine kursive Serifen-Überschrift als Abkürzung zum edlen Look.
- Größen als Skala mit clamp(): Fließtext 16 bis 18 px, Zeilenhöhe 1,5 bis 1,7; Überschriften 1,05 bis 1,2. H1, H2 und H3 unterscheiden sich deutlich.
- Zeilenlänge 60 bis 75 Zeichen. text-wrap: balance für Überschriften, pretty für Absätze. Tabellarische Ziffern bei Preisen und Zahlen.

Abstände: 4-px-Raster mit der Skala 4, 8, 12, 16, 24, 32, 48, 64, 96, 128. Was zusammengehört, steht eng, Abschnitte stehen weit auseinander. Seitenrand am Handy mind. 16 px. Textbreite höchstens etwa 680 px, Inhalt höchstens etwa 1200 px.

```
Radien: höchstens drei Werte, passend zur Wirkung (eckiger wirkt präziser, runder freundlicher).
Flächen trennen sich über Abstand, Linie oder Tönung. Schatten nur für Ebenen über der Seite (Menü, Dialog, Banner), höchstens zwei Stufen.
Bewegung: 120 ms für Rückmeldungen, 200 ms für Zustände, 320 ms für größere Ein- und Ausblendungen. Eintreten mit ease-out, Austreten mit ease-in. Kein Federn, kein Überschwingen.
Breakpoints: Basis 360 px, dann 768, 1024, 1280 (in DESIGN.md, Media Queries können keine Variablen lesen). Ebenen (z-index) als Variablen: Header, Menü, Banner, Dialog, Meldung.

Wirkung nach Branche:
- Arzt-, Zahnarzt-, Therapiepraxis: ruhig und nahbar. Ruhige Display-Schrift (Serife mit wenig Kontrast oder humanistische Grotesk), offene Textschrift. Helle, kühle oder grüne Töne, gedämpfter Akzent, viel Luft, weiche Radien. Kein Krankenhausblau, keine Zahn- oder Herz-Icons.
- Steuerberatung, Kanzlei, Finanzen: kompetent und präzise. Display mit Haltung (klassische Serife oder strenge Grotesk), Textschrift mit tabellarischen Ziffern. Tiefe Töne wie Petrol, Tannengrün, Marine oder Anthrazit auf hellem Grund, ein klarer Akzent, kleine Radien, strenges Raster. Kein Händeschütteln, keine Taschenrechner.
- Marketing, Agentur, Kreative: kreativ und trotzdem kompetent. Markante Display-Schrift (breit, schmal oder mit Eigenheiten), ruhige Textschrift. Ein mutiger Akzent auf ruhiger Basis, große Typografie, Arbeitsproben im Mittelpunkt, klares Raster, damit es nicht beliebig wirkt.
- Handwerk, Bau: robust und direkt. Kräftige Grotesk, erdige Töne oder eine Signalfarbe, große Projektfotos, Telefonnummer immer sichtbar.
- Gastronomie, Hotel: warm und einladend. Display mit Persönlichkeit, große Fotos von Essen und Räumen, Speisekarte als Text statt PDF, Reservierung im ersten Bildschirm.
- Andere Branche: leite die Wirkung aus Zielgruppe und Angebot ab und schreib sie in DESIGN.md.
Man soll die Branche ohne Logo erkennen, ohne dass die Seite wie die der Konkurrenz aussieht.


5. KEINE TYPISCHEN KI-DESIGNS

Lass diese Muster weg, außer ich verlange eines ausdrücklich:
- lila-blaue Verläufe, Text mit Farbverlauf, Neon auf dunklem Grund, Lichthöfe hinter Inhalten
- Glas-Karten, Karten in Karten, Raster aus gleichen Karten mit Icon in einer runden Kachel
- cremefarbener Hintergrund mit Serifen-Überschrift und Terrakotta-Akzent
- kleine Etiketten in Großbuchstaben über Überschriften (Kicker), Pillen-Badge über der Hero-Überschrift
- riesige Kennzahlen mit kleinem Label ohne echte Quelle
- farbiger Streifen an einer Kartenseite, derselbe Radius und Schatten auf allem
- Elemente, die hüpfen, wackeln, pulsieren, Laufbänder, blinkender Cursor
- Emojis als Icons, ein Pfeil hinter jedem Button-Text, Mittelpunkte zwischen Stichwörtern
- erfundene Bewertungen, „Bekannt aus“-Logos, Stockfotos, KI-generierte Menschen
- überall gleiche Abstände, alles zentriert
Nutze beim Bau den Skill impeccable. Lass am Ende npx impeccable detect über die fertige Seite laufen und behebe jeden Fund oder begründe in DESIGN.md, warum er bleibt.


6. TEXTE

- Schreib die Texte aus meinen Angaben. Erfinde keine Fakten, Preise, Zahlen, Namen oder
```

Bewertungen. Fehlt etwas, setz [Platzhalter] und liste ihn am Ende auf.
- Lass jeden Text durch den Skill humanizer laufen. Lies danach auf Deutsch nach: keine Floskeln wie „In der heutigen Zeit“, „maßgeschneidert“, „ganzheitlich“, „Ihr Partner für“, kein „nicht nur, sondern auch“, keine Dreierketten aus Gewohnheit, keine Gedankenstriche als Satzverbinder.
- Direkt zum Punkt. Kein Vorwort, keine Erklärung, die der Besucher nicht braucht. Absätze höchstens vier Zeilen am Handy, Sätze meist unter 20 Wörtern.
- Überschriften sagen die Aussage, nicht das Thema: „Angstpatienten behandeln wir ohne Zeitdruck“ statt „Unsere Leistungen“. Keine Kicker-Texte.
- Zahlen und Beispiele statt Adjektive.
- Fehlermeldungen sagen, was falsch ist und wie es richtig geht: „Bitte gib eine E-Mail-Adresse mit @ ein.“ statt „Ungültige Eingabe“.

7. BEWEGUNG UND DETAILS

Bewegung zeigt, was gerade passiert. Sie bleibt klein und schnell.
- Buttons: Hover etwas dunkler (120 ms), beim Drücken minimal kleiner (scale 0.98), Fokus-Ring nur bei Tastatur (:focus-visible), deaktiviert erkennbar. Ladezustand: Button sofort gesperrt, Text „Wird gesendet …“, ein Spinner im Button erst, wenn es länger als eine Sekunde dauert. Die Breite bleibt gleich.
- Links im Fließtext bleiben unterstrichen, beim Hover ändern sich Farbe oder Strichstärke.
- Formulare: prüfen beim Verlassen des Felds, nicht bei jedem Tastendruck. Fehler steht unter dem Feld mit Text und Icon, nie nur als rote Farbe. Beim Absenden springt der Fokus zum ersten Fehler. Erfolg mit klarer Meldung oder Danke-Seite. Doppeltes Absenden geht nicht.
- Abbruch beim Senden (kein Netz, Serverfehler, Zeitüberschreitung): Alle Eingaben bleiben stehen, die Meldung sagt, was passiert ist, und bietet „Erneut senden“ und die Telefonnummer an.
- Seitenwechsel: leise Überblendung über die View Transitions API, 150 bis 250 ms.
- Beim ersten Laden blenden die Elemente des ersten Bildschirms einmal gestaffelt ein (Deckkraft und 8 bis 12 px, je 40 bis 60 ms versetzt). Nicht jeder Abschnitt beim Scrollen. Ohne JavaScript ist alles sichtbar.
- Ladeanzeigen: unter einer Sekunde keine, bei Aktionen bis wenige Sekunden ein Spinner, für Bereiche mit nachladendem Inhalt ein Skeleton in der Form des Inhalts, über zehn Sekunden ein Fortschrittsbalken.
- Bilder mit festem Seitenverhältnis (nichts springt), Platzhalter in Flächenfarbe, weiches Einblenden nach dem Laden.
- Meldungen (Toasts) mit aria-live, verschwinden nach 4 bis 6 Sekunden.
- Hover-Effekte nur bei Geräten mit Maus (@media (hover: hover)). Bewegt wird nur über transform und opacity, Farbwechsel bei Hover und Fokus sind erlaubt. Bei prefers-reduced-motion fällt jede Bewegung weg.

8. NAVIGATION

- Header: Logo (führt zur Startseite), die vier bis sechs wichtigsten Seiten, die Hauptaktion als Button. Die aktuelle Seite ist markiert (aria-current).
- Sticky nur bei langen Seiten: beim Runterscrollen ausblenden, beim Hochscrollen wieder zeigen. Sonst statisch. Entscheide und begründe es in DESIGN.md.
- Handy: Menü-Button mind. 44 × 44 px, das Menü öffnet in unter 250 ms, das Icon wird zum X. Drin: alle Seiten, die Hauptaktion, die Telefonnummer. Die Seite dahinter scrollt nicht mit, Escape und Tippen daneben schließen, der Fokus bleibt im Menü und springt danach zurück zum Button. Es funktioniert sofort nach dem Laden.
- Footer: Kontakt, Öffnungszeiten, alle Seiten, Impressum, Datenschutz, AGB (falls

```
vorhanden), Cookie-Einstellungen (sobald ein Dienst eine Einwilligung braucht).
- Am Handy optional eine feste Leiste unten mit Anrufen und Hauptaktion, wenn das zum Ziel
passt.


9. BILDER UND ICONS

- Echte Fotos aus meinem Ordner: Team, Räume, Arbeit, Ergebnisse. Keine Stockfotos, keine
KI-Menschen.
- Fehlen Fotos, setz gestaltete Platzhalter im Design-Set (Fläche mit kurzer Beschreibung
des Motivs) und gib mir eine Fotoliste: Motiv, Format, Seite.
- Bilder als AVIF oder WebP mit srcset, width und height. Ab dem zweiten Bildschirm lazy,
das erste große Bild mit fetchpriority="high". Alt-Texte konkret, dekorative Bilder mit
alt="".
- Logo als SVG, Favicon als SVG plus 180-px-Icon für Apple, OG-Bild 1200 × 630 px.
- Icons aus einer Familie, als Inline-SVG, Strichstärke passend zur Textschrift, gleiche
Größe. Nur wo sie etwas erklären (Telefon, Uhrzeit, Ort). Keine Icon-Kacheln über
Überschriften, keine Emojis.


10. BARRIEREFREIHEIT (WCAG 2.2 AA)

- Semantisches HTML (header, nav, main, footer), eine H1 pro Seite, keine übersprungenen
Ebenen, Link „Zum Inhalt springen“.
- Kontrast: Text 4,5:1, große Schrift und Bedienelemente 3:1, auch bei Hover, Fokus und
Fehler.
- Alles per Tastatur bedienbar, logische Reihenfolge, Fokus immer sichtbar.
- Tippflächen mind. 44 × 44 px mit Abstand dazwischen.
- Formulare mit sichtbaren Labels (Platzhalter ist kein Label), Pflichtfelder markiert,
Fehler per aria-describedby am Feld, Statusmeldungen per aria-live.
- lang="de", Zoom bis 200 % ohne Verlust, keine Information nur über Farbe.
- Verkaufe ich online an Verbraucher, gilt seit 28.06.2025 das
Barrierefreiheitsstärkungsgesetz (Ausnahme: Kleinstunternehmen mit Dienstleistungen). Dann
sag es mir und leg die Seite mit den Informationen zur Barrierefreiheit an.


11. MOBILE FIRST

- Erst für 360 bis 390 px bauen, dann erweitern. Bei 320 px kein seitliches Scrollen.
- Erster Bildschirm am Handy: was es ist, für wen, Hauptaktion. Ohne Scrollen.
- Hauptaktion mit dem Daumen erreichbar, Telefonnummer als tel:-Link, Adresse als Link zur
Karten-App.


12. COOKIE-BANNER UND RECHTSTEXTE

- Bau den Einwilligungs-Banner immer ein, gestaltet im Design-Set. Er fragt genau die
Dienste ab, die ich in Abschnitt 1 angekreuzt habe.
- Vor dem Ja lädt kein Dienst: keine Skripte, Pixel, Karten, Videos oder Schriften von
Dritten. Google Maps und YouTube stehen bis dahin als Platzhalter mit Button „Karte laden“
bzw. „Video laden“. Google Analytics und Google Ads laufen mit Consent Mode v2, Standard
„denied“.
- „Alle ablehnen“ steht gleich groß und auf derselben Ebene wie „Alle akzeptieren“, dazu
„Einstellungen“ mit einem Schalter je Dienst. Nichts ist vorausgewählt. Impressum und
Datenschutz bleiben erreichbar.
```

- Gespeichert wird nur die Wahl. Widerruf jederzeit über „Cookie-Einstellungen“ im Footer, dabei löschst du die Cookies, die die Seite selbst setzen kann, und lädst die Dienste nicht mehr.
- Lädt die Seite keinen Dienst, der eine Einwilligung braucht, ist der Banner vorbereitet, aber aus. Dann sag es mir.
- Impressum nach § 5 DDG (bei Kammerberufen mit Kammer, Berufsbezeichnung und berufsrechtlichen Regeln). Datenschutzerklärung nach Art. 13 DSGVO: Hoster, Server-Logs, Kontaktformular, E-Mail und jeder Dienst mit Anbieter, Zweck, Rechtsgrundlage, Speicherdauer und Übermittlung in Drittländer, dazu die Rechte der Besucher. AGB und Widerrufsbelehrung nur bei Verkauf. Alles im Footer verlinkt.
- Markier die Rechtstexte als Entwurf und sag mir, dass ich sie von einem Anwalt oder mit einem Generator wie eRecht24 prüfen lassen muss.

13. SEO UND SOCIAL SHARING

- Je Seite: eigener Title (höchstens 60 Zeichen: Leistung, Ort, Marke), Meta-Description (140 bis 160 Zeichen, mit Nutzen und Aufforderung), eine H1, sprechende URL (/leistungen/implantate/), Canonical.
- Transaktionale Suchbegriffe: Wonach sucht jemand, der jetzt buchen oder kaufen will? Zum Beispiel „Zahnarzt Köln Ehrenfeld Termin“ oder „Steuerberater für Selbstständige München“. Leg je Seite einen Hauptbegriff fest (Liste in SEO.md) und bring ihn in Title, H1, URL und einen Alt-Text, im Text nur dort, wo er natürlich passt. Kein Keyword-Stopfen.
- Open Graph und Twitter-Card je Seite: og:title, og:description, og:image (1200 × 630 px), og:url, og:type, og:locale de_DE.
- JSON-LD: LocalBusiness oder der passende Untertyp (Dentist, AccountingService, ProfessionalService …) mit Adresse, Telefon, Öffnungszeiten und Geo-Daten, dazu BreadcrumbList. FAQPage nur bei sichtbarer FAQ, keine erfundenen Bewertungen im Markup.
- sitemap.xml, robots.txt, 301-Weiterleitungen von Adressen der alten Website.
- Jede Leistungsseite verlinkt den Kontakt und verwandte Leistungen.
- Tempo: LCP unter 2,5 s, CLS unter 0,1, INP unter 200 ms. Die wichtigste Schrift per preload, font-display: swap, kein unnötiges JavaScript.
- Erinnere mich am Ende an das Google-Unternehmensprofil mit derselben Adresse und Telefonnummer wie auf der Seite.

14. FORMULARE

- Steht oben keine E-Mail-Adresse für Anfragen, frag danach.
- Sag mir vorher klar: Eine rein statische Seite kann ein Formular nicht selbst verschicken, dafür braucht es einen Server. Schlag mir je nach Hosting den Weg vor:
  - Vercel oder Netlify: eine Serverfunktion, die über das SMTP meines Postfachs oder einen Maildienst sendet. Zugangsdaten nur als Umgebungsvariable.
  - Webhosting bei Strato oder IONOS: ein PHP-Skript, das über das SMTP meines Postfachs sendet.
  - Sonst: ein Formulardienst mit Servern in der EU und Vertrag zur Auftragsverarbeitung.
- Nur die Felder, die gebraucht werden. Datenschutzhinweis mit Link direkt über dem Button, keine vorausgewählte Checkbox.
- Spam-Schutz ohne reCAPTCHA: verstecktes Honeypot-Feld und Zeitprüfung auf dem Server.
- Danke-Seite immer, Bestätigungsmail an den Absender nur, wenn ich es will.
- Test: Eine echte Testanfrage kommt an. Ist der Server weg, erscheint die Fehlermeldung aus Abschnitt 7.

15. PRÜFEN, BEVOR DU „FERTIG“ SAGST

- Build ohne Fehler, keine Fehler in der Konsole, alle Links antworten mit 200.
- Screenshots jeder Seite bei 375, 768 und 1440 px. Sieh sie dir an und behebe, was nicht stimmt.
- Lighthouse am Handy: Performance, Barrierefreiheit, Best Practices und SEO je mind. 90 (SEO nicht bei Danke- und 404-Seite, die sind noindex).
- Ohne Einwilligung geht keine Anfrage an Dritte (Netzwerk-Tab).
- npx impeccable detect über die fertige Seite, Funde beheben.
- Gib mir am Ende eine Liste: jede Seite mit Ziel und Hauptbegriff, offene Platzhalter, Fotoliste, was ich rechtlich prüfen lassen muss.

16. VERÖFFENTLICHEN

Frag mich zum Schluss:
1. Auf welcher Domain soll die Seite laufen, und wo liegt die Domain (z. B. Strato, IONOS)?
2. Bei welchem Anbieter soll die Seite laufen?
Führ mich dann Schritt für Schritt durch, ein Schritt pro Nachricht, und warte nach jedem auf mein „erledigt“:
- Next.js auf Vercel: Projekt auf GitHub hochladen, bei vercel.com mit GitHub anmelden, Projekt importieren, Umgebungsvariablen eintragen (z. B. für das Formular), deployen. Unter Settings › Domains die Domain hinzufügen. Nenn mir genau die DNS-Einträge, die Vercel dort anzeigt, und wo ich sie bei meinem Domain-Anbieter eintrage. Die Nameserver bleiben, wo sie sind, die MX-Einträge für E-Mail fasst du nicht an. HTTPS richtet Vercel selbst ein.
- Statische Seite bei Strato oder IONOS: Es braucht ein Webhosting-Paket, eine Domain allein reicht nicht. Build erzeugen, per SFTP in das Verzeichnis laden, auf das die Domain zeigt, SSL im Kundenbereich aktivieren, HTTP auf HTTPS umleiten.
- Anderer Anbieter: frag, was ich gebucht habe, und erklär es genauso.
Danach: Seite live öffnen, Formular und Cookie-Banner testen, Sitemap in der Google Search Console einreichen.

So setzt du ihn in Claude Code ein

1 Installier impeccable und den Humanizer

Beide sind kostenlos. impeccable kennt die typischen KI-Designmuster, der Humanizer nimmt deinen Texten den KI-Ton. Führ die zwei Zeilen im Projektordner aus oder lass Claude Code das machen.

Installieren

```
npx impeccable install -y
npx skills add blader/humanizer --agent claude-code -y
```

2 Füll Abschnitt 1 aus

Firma, Leistungen mit Preisen, Zielgruppe, Kontakt, welche Dienste du nutzt und an welche Adresse Anfragen gehen. Leg echte Fotos in einen Ordner im Projekt. Rechne mit etwa 20 Minuten. Was du nicht weißt, lässt du leer, Claude fragt nach.

3 Schick ihn ab

Claude stellt zuerst alle offenen Fragen auf einmal, legt das Design-Set an und baut dann alle Seiten in einem Rutsch. Zum Schluss prüft es mit impeccable und fragt dich, auf welcher Domain und bei welchem Anbieter die Seite laufen soll.

MEIN TIPP

Ändern immer im Design-Set

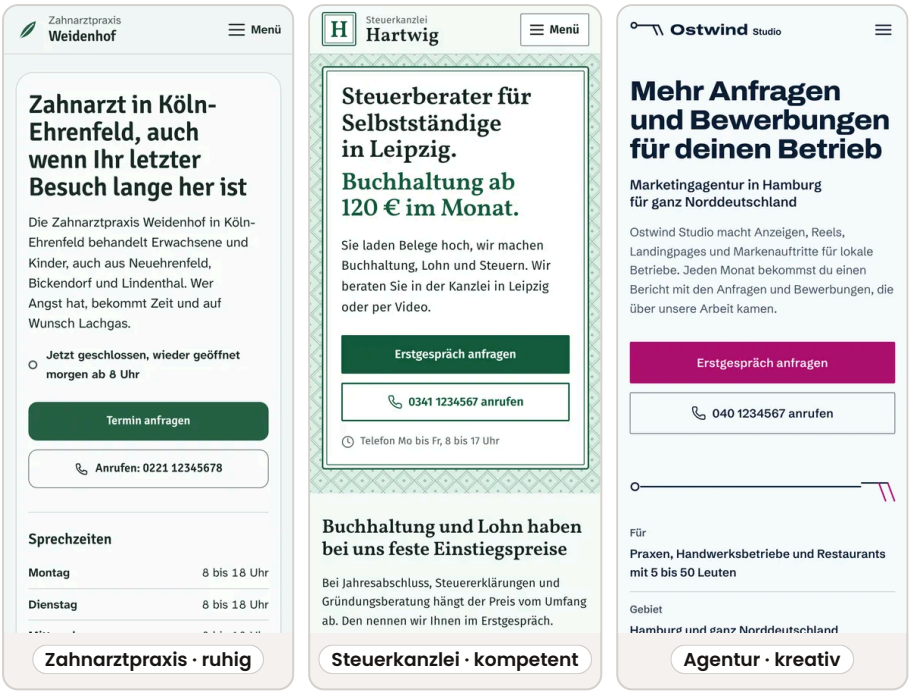
Willst du danach etwas anders haben, sag Claude, es soll das im Design-Set ändern. Änderst du Seite für Seite per Prompt, passen Abstände und Farben bald nicht mehr zusammen.

Du willst wissen, wo KI in deinem Betrieb Zeit spart? Buche einen Call mit mir und wir finden es gemeinsam raus.

Call buchen

Die Branche bestimmt Schrift und Farbe

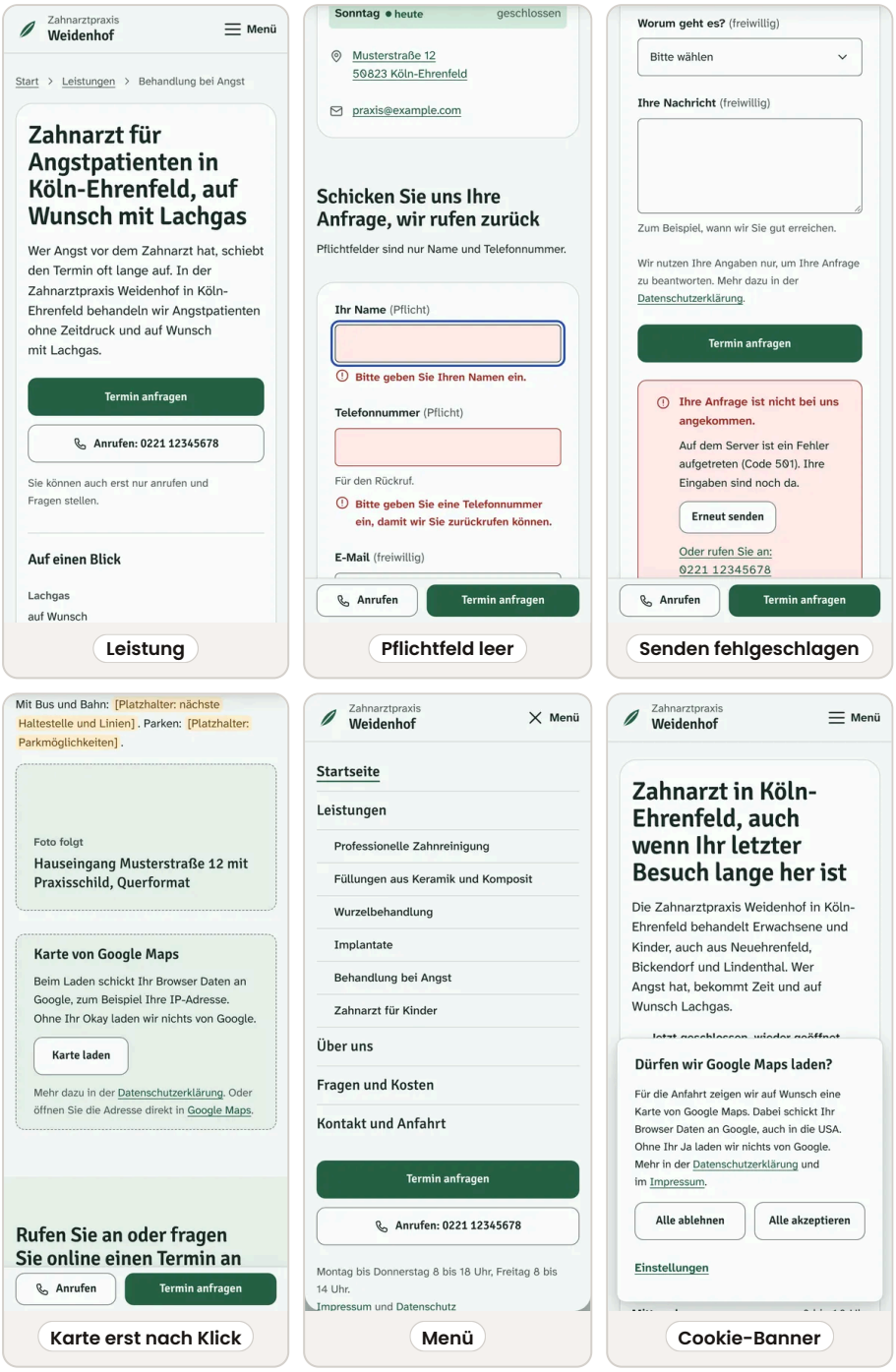
Im Prompt steht, wie eine Praxis, eine Kanzlei oder eine Agentur wirken soll. Claude wählt Schriften und Farben danach.



Derselbe Prompt, drei erfundene Betriebe.

Alle Seiten kommen aus einem Design-Set

Die Beispielpraxis von oben, alles aus einem Durchgang: 15 Seiten mit denselben Abständen, Farben und Schriften.





Automatisierungs-Call für Agenturen, Coaches und Dienstleister

In 30 Minuten schauen wir gemeinsam, wo KI-Agenten und automatisierte Prozesse in deinem Betrieb zuerst Zeit und Geld sparen. Kostenlos.

FÜR WEN IST DAS?

- ✓ Agenturen, Coaches und Dienstleister
- ✓ ab 10.000 € Monatsumsatz
- ✓ ein klarer Flaschenhals im Tagesgeschäft
- ✓ du willst KI im Betrieb einsetzen

Call buchen

api.leadconnectorhq.com/widget/booking/mXsszCow3Sxqm4QEnOxK

Scannen & Termin wählen