

HAPPY BIRTHDAY, SASKIA!



Willkommen in deiner
Claude-Stadt!

NotFall

**Claude
anschreien?**

Erst hier
drücken

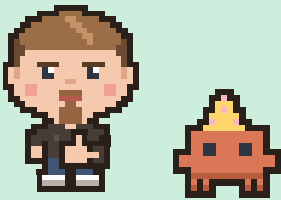


Zum Mitnehmen

Dein Claude-Buddy fürs eigene Claude.

Alles liegt in einem Repo: der Laurin-Helfer als Skill und Pixel-Laurin, der dir auf dem Bildschirm zeigt, wo du klicken musst.

<https://github.com/Idefixart/saskia>



So installierst du ihn

- 1 **Pixel-Laurin auf deinen Bildschirm:** `laurin-buddy.mcpb` laden und doppelklicken. Claude Desktop installiert ihn. Wenn Claude fragt, die Werkzeuge erlauben.
- 2 **Laurin-Helfer in die Desktop-App:** Settings › Capabilities › „Code execution and file creation“ an, dann Customize › Skills › „+“ › „Upload a skill“ › `laurin-helfer.zip`.
- 3 **In Antigravity:** im Terminal einfügen:

```
npx skills add Idefixart/saskia -a antigravity -g -y --  
copy
```

- 4 **Ausprobieren:** „Zeig mir, wo ich in Shopify ein Produkt anlege.“

Post von Laurin



Liebe Saskia,

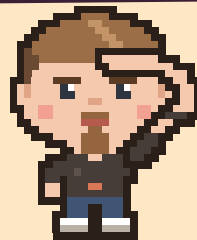
alles Liebe zum Geburtstag! Wenn du das hier liest, hast du gerade dein Handy an den Anhänger gehalten: Willkommen in deiner Claude-Stadt. Clawd und ich wohnen ab jetzt an deinem Schlüsselbund. Wann immer Claude mal wieder nicht kapiert, was du willst, einfach wieder dranhalten.

Bevor du ihn das nächste Mal anschreiest, drück lieber unseren Notfallknopf, der steckt das deutlich besser weg.

Und weil ich dich mag, sag ich's hier noch mal ganz sanft:

Shops baust du ab jetzt mit Shopify, nicht selbst. Lass dich heute ordentlich feiern!

Dein Laurin



Stadtplan

Wo willst du hin?

Elf Orte, alle zu Fuß erreichbar. Tipp einfach drauf.

01

Werkzeug-Garage

Welches Tool wofür

02

Notaufnahme

Der Anschrei-
Notfallknopf

03

Regel-Boulevard

Sieben goldene Regeln

04

Brief-Werkstatt

Prompt-Baukasten

05

Baustellen-Route

Von der Idee bis live

06

Shopify-Mall

Shops? Shopify.

07

Bibliothek

Laurins Skills

08

Superkräfte-Labor

Was Claude neu kann

09

Praxis Dr. Fehler

Symptom rein, Fix raus

10

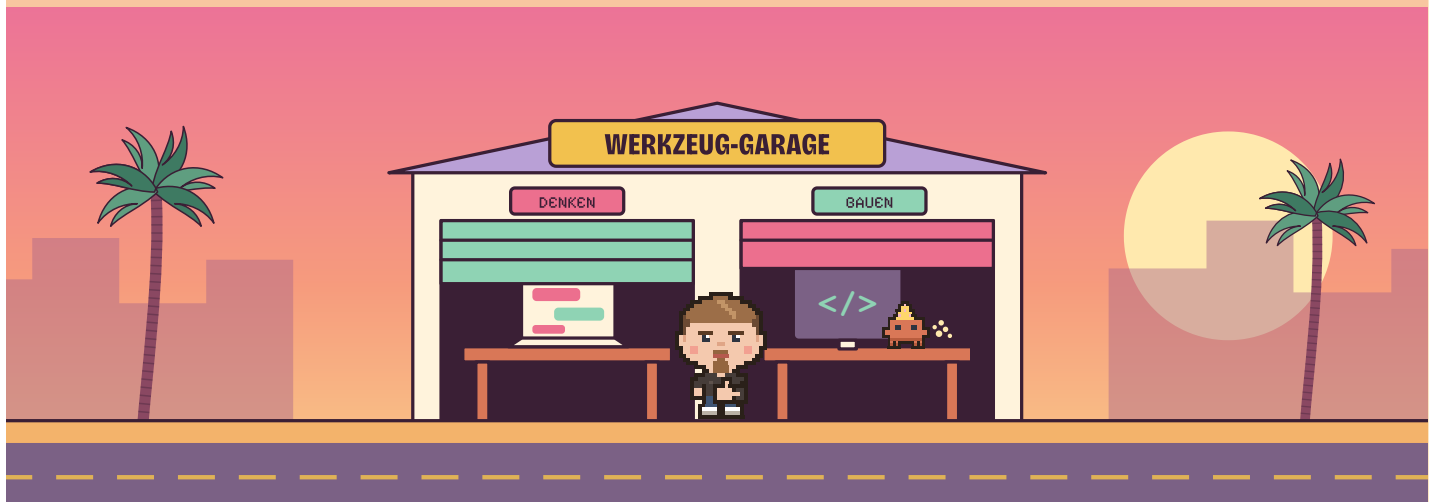
Buddy-Hangar

Pixel-Laurin auf deinem
Screen

11

Kiosk

Alles zum Runterladen



Denken in der App, bauen in Antigravity.

Zwei Werkzeuge, zwei Jobs. Wer im Chat Code baut oder in der IDE Texte grubelt, macht sich das Leben schwer.

Claude-Desktop-App

Zum Denken

- Briefs, Texte, Pläne: im Chat oder in einem Projekt mit Anweisungen und Wissensdateien.
- Recherche mit `/deep-research` oder „+“ › Research. Bezahlte Pläne, frisst Limit.
- Design-Entwürfe mit Claude Design, direkt im Chat oder im Artifacts-Tab. Export als HTML oder ZIP.
- Skills unter Customize › Skills, Connectors wie Shopify unter Customize › Connectors.
- Code-Tab: Claude Code ohne Terminal, mit Browser-Vorschau deiner Seite.
- Hier wohnt auch Pixel-Laurin als Zeige-Buddy.

Antigravity mit Claude

Zum Bauen

- Code, Dateien, ganze Projektordner.
- Claude Opus 5.5 und Sonnet 5.5 (Thinking) im Dropdown unter dem Eingabefeld. Braucht Google AI Pro, Ultra oder Enterprise.
- Regeln in `.agents/rules/`, mit `trigger`-Zeile oben, sonst werden sie ignoriert.
- Skills mit `npx skills add <repo> -a antigravity -g -y`, auf Windows plus `--copy`.
- Plan B: die offizielle Erweiterung „Claude Code“ aus dem Extensions-Panel, läuft über dein Claude-Abo.
- Keine Workflows mehr bauen: Die hören am 19.10.2026 auf.

Welches Modell wann?

Sonnet 5.5

Alltag, Bauen, Routine. Schont das Limit.

Opus 5.5

Knifflige Bugs, große Pläne, Research.

Haiku 5.5

Kleinkram: Texte umformulieren, kurze Fragen.

Fable 5.1

Auf Max dabei, auf Pro nur über Usage-Credits. Frisst viel.

Effort (Low bis Max) findest du im Menü neben dem Senden-Knopf: Low für Kleinkram, High für harte Nüsse.



02 · Notaufnahme

Erst drücken, dann tippen.

Großbuchstaben und Schimpfwörter enthalten keine Information. Claude braucht den Unterschied zwischen dem, was er gemacht hat, und dem, was du wolltest.



Und jetzt das hier in einen neuen Chat:

Stopp, wir machen einen Reset.

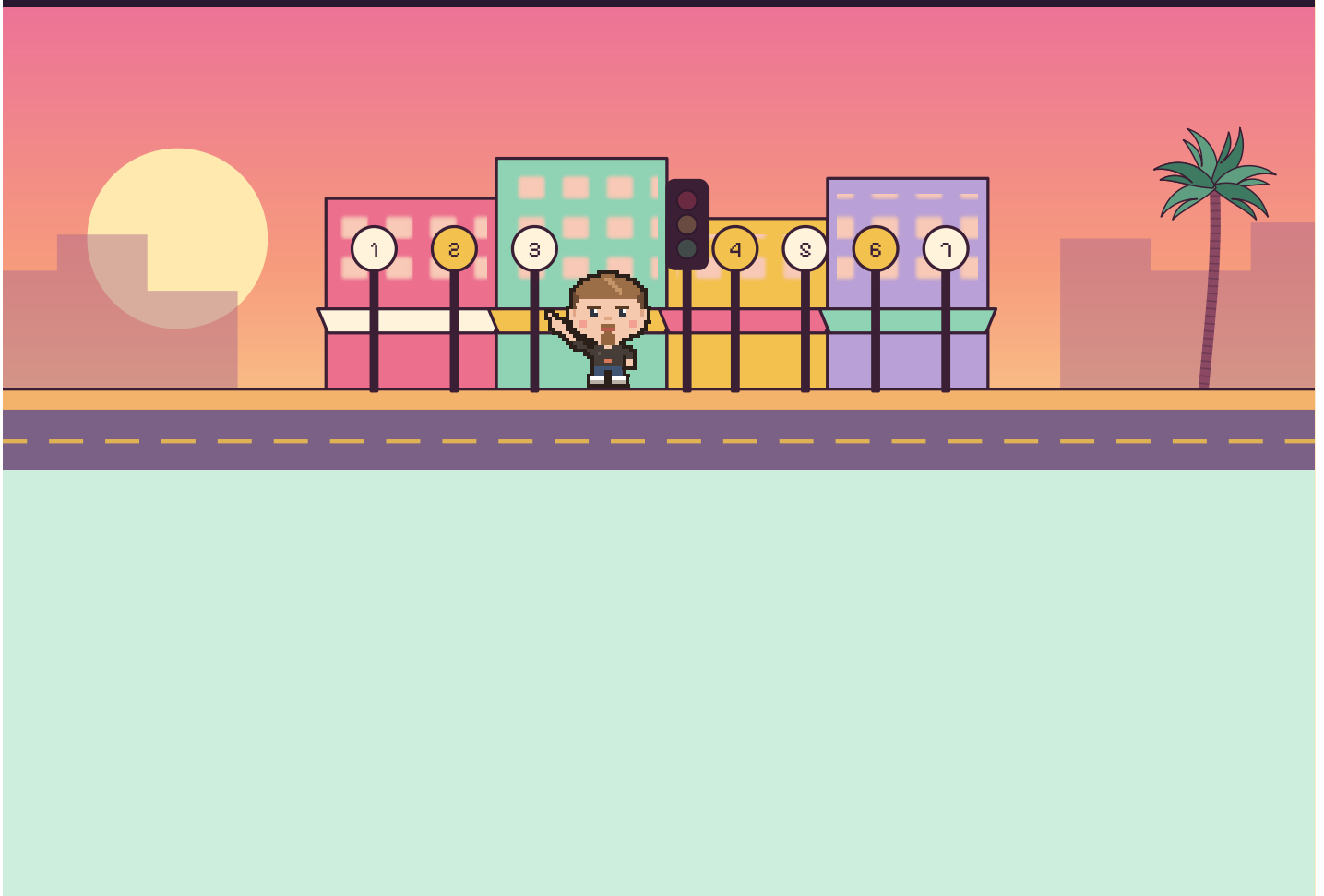
Ziel: [was am Ende funktionieren soll]

Was passiert ist: [was du gemacht hast / was ich gerade sehe]

Was ich stattdessen will: [genau beschreiben, gern mit Screenshot]

Schon versucht: [was nicht geklappt hat]

Bitte noch nichts ändern. Nenn mir zuerst die wahrscheinlichste Ursache und zeig mir einen kurzen Plan. Ich sag dann Okay.



Sieben Gewohnheiten, die mehr bringen als jeder Wutanfall.

1

Erst planen

Lass Claude zuerst einen Plan zeigen und gib dein Okay, bevor Code entsteht.

Fünf Minuten Plan sparen zwei Stunden Rückbau.



2

Eins nach dem anderen

Eine Section, ein Bug, ein Wunsch pro Nachricht.

Drei Wünsche auf einmal? Ich bin Pixel-Laurin, kein Flaschengeist.

3

Zeigen statt beschreiben

Screenshots und Referenz-Links sagen mehr als jedes Adjektiv.

„Moderner“ heißt für jeden was anderes. Ein Link nicht.

4

Neuer Chat pro Thema

Lange Chats machen Claude vergesslich und fressen dein Limit.

Zusammenfassen lassen, neu starten, weiter geht's.



Anweisungen pflegen

Was immer gilt, gehört in die Projekt-Anweisungen oder die Antigravity-Regeln.

Einmal aufschreiben, nie wieder erklären.



Ursache vor Rumprobieren

Nach zwei Fehlversuchen erst die Ursache suchen lassen, dann fixen.

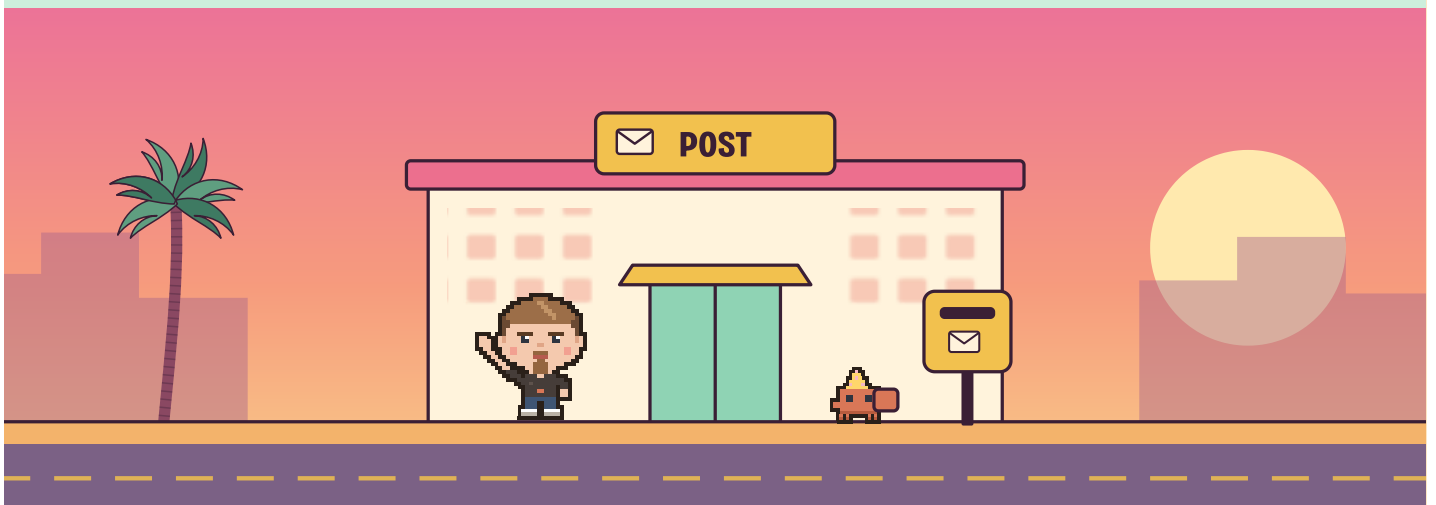
Raten ist teuer. Nachdenken ist billig.



Prüfen lassen

Fertig ist es erst, wenn es im Browser und am Handy geprüft ist.

„Fertig“ heißt angeschaut, nicht nur behauptet.



Formular ausfüllen, fertigen Brief kopieren.

Ein guter Brief ist die halbe Website. Alles bleibt in deinem Browser, nichts wird gespeichert oder verschickt.

Ich möchte eine Website bauen. Bitte lies den Brief und zeig mir zuerst einen Plan. Noch keinen Code.

Name/Marke: [noch offen]

Ziel: [noch offen]

Zielgruppe: [noch offen]

Vibe: [noch offen]

Referenzen: [noch offen]

Seiten: [noch offen]

Must-haves: [noch offen]

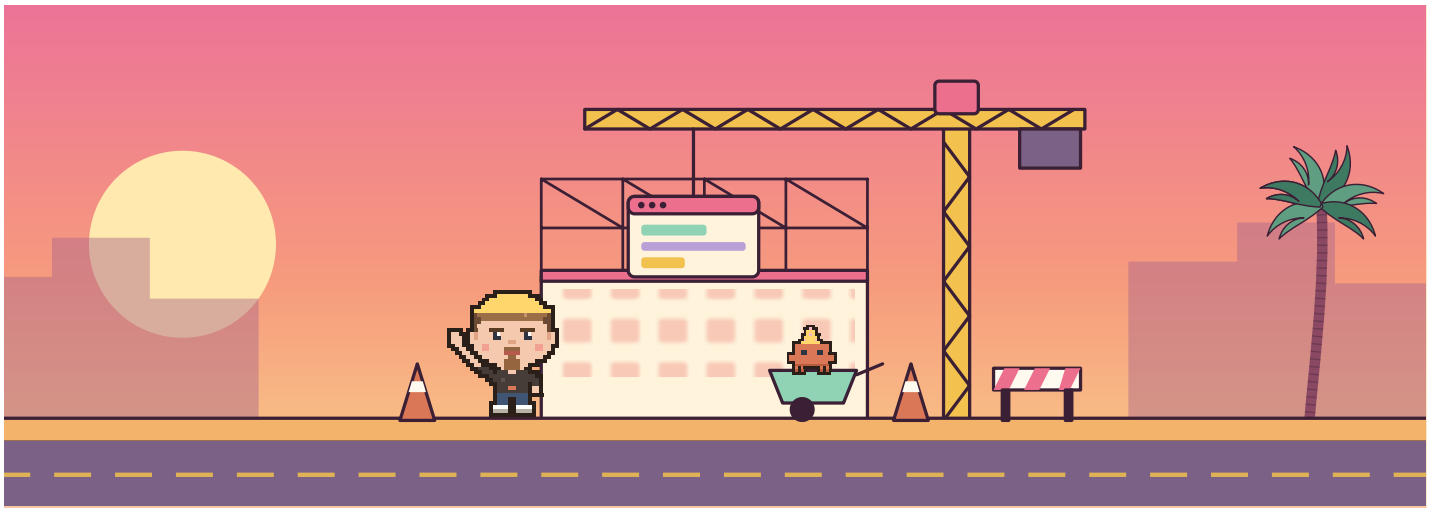
No-Gos: [noch offen]

Rechtliches: Impressum und Datenschutz

So arbeiten wir:

1. Stell mir maximal drei Rückfragen, wenn etwas unklar ist.
2. Schlag mir Farben, Schriften und Abstände als DESIGN.md vor.
3. Bau mobil zuerst, Section für Section, und zeig mir jeden Schritt.
4. Keine KI-Standards: keine lila Verläufe, kein Hero mit drei gleichen Karten, keine Emojis als Aufzählung.
5. Fertig ist es erst, wenn du es im Browser auf Handy und Desktop geprüft hast.

Tipp: Brief in einen neuen Chat, erst den Plan abnicken, dann bauen lassen.



09 · Baustellen-Route

Acht Stationen. Immer in dieser Reihenfolge.

1

Brief

Ziel, Zielgruppe, Vibe, Seiten und No-Gos einmal sauber aufschreiben.

2

Referenzen

Drei bis fünf echte Seiten, die du magst, und was genau daran.

3

Design festlegen

Farben, Schriften und Abstände in eine DESIGN.md, bevor gebaut wird.

4

Bauen

Section für Section, mobil zuerst, nach jedem Schritt anschauen.

5

Polieren

critique, polish, typeset, animate, adapt, optimize. Eins pro Runde.

6

Testen

Handy, Tablet, Desktop, hell und dunkel, im echten Browser.

7

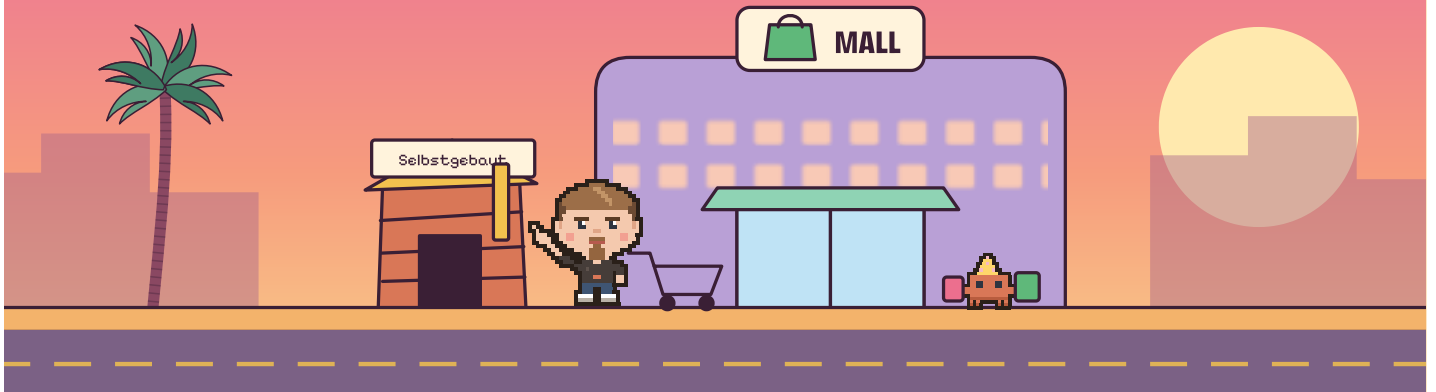
Rechtliches

Impressum und Datenschutz, beim Shop dazu AGB, Widerruf und Versand.

8

Live

In einem Rutsch live schalten und auf der echten Adresse am Handy prüfen.



Shops? Shopify.

Laurin sagt: Selbstgebaute Shops über Claude sind Quatsch. Claude kann dir einen Warenkorb bauen. Aber Checkout, Zahlungen, Steuern und Sicherheit sind kein Wochenendprojekt.

Selbstgebaut mit Claude	vs	Shopify
Claude baut dir einen Checkout. Ob der sicher ist, merkst du beim ersten Kunden.	✓	Den Checkout pflegt Shopify, sicher nach PCI-DSS Level 1.
Server, Zertifikat, Backups: alles deine Baustelle.	✓	Hosting und SSL sind inklusive.
Zahlungsanbieter anbinden heißt: ein Wochenende API-Doku.	✓	Shopify Payments: 2,1 % + 0,30 € pro Zahlung (Basic), Klarna 2,99 % + 0,35 €.
Steuersätze pflegst du von Hand und hoffst.	✓	Die MwSt. rechnet Shopify automatisch. Abführen bleibt deine Sache.
Widerruf, AGB, Versand: abtippen und Daumen drücken.	✓	Die Seiten werden generiert. Deutsche Texte am besten aus einer Kanzlei-App, das Impressum machst du selbst.
Claude rät, wie dein eigener Code gemeint war.	✓	Shopify AI Toolkit und Dev MCP geben Claude die echte Shopify-Doku.
Kostet „nichts“. Außer deine Abende.	✓	Basic: 36 € im Monat, 27 € bei jährlicher Zahlung. Testen: 3 Tage, dann 1 € im Monat für 3 Monate.
Preise Deutschland, Stand Oktober 2026. Fremde Zahlungsanbieter kosten auf Basic 2 % extra. OSS wird ab 10.000 € EU-Fernverkauf Thema.		

Sicher mit Claude am Theme arbeiten

- 1 **Dev-Store nutzen.** Nie am echten Shop üben.
- 2 **Theme holen:** `shopify theme pull`
- 3 **Lokal entwickeln:** `shopify theme dev --store <shop>` startet ein verstecktes Entwicklungs-Theme mit Live-Reload.
- 4 **Prüfen und Vorschau:** Theme Check, dann `shopify theme push --unpublished`.
- 5 **Live schaltest du selbst im Admin.** Nie `--live`, `--publish` oder `--allow-live`.

Zum Kopieren

Wir arbeiten an meinem Shopify-Theme. Regeln für diese Sitzung:

- Nur Dev-Store oder unveröffentlichtes Theme.
- Erlaubt: `shopify theme pull`, `shopify theme dev`, `shopify theme check`,
`shopify theme push --unpublished`.
- Verboten: `--live`, `--publish`, `--allow-live`. Live schalte ich selbst im Admin.
- Bevor du einen Befehl ausführst: zeig ihn mir und sag in einem Satz, was er tut.

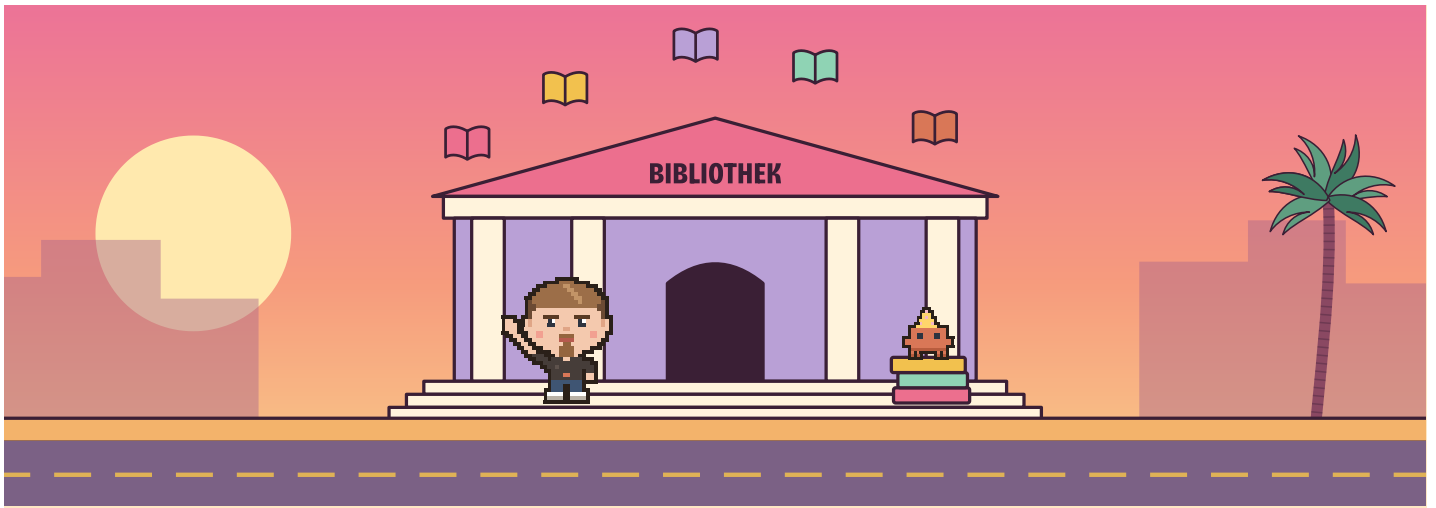
Erste Aufgabe: [...]

Bau mir eine neue Section für mein Theme: [was sie zeigen soll]. Halte dich an die Theme-Standards (Liquid, Barrierefreiheit) und prüf

den Liquid-Code mit dem Shopify AI Toolkit.

Texte, Bild und Farben sollen im Theme-Editor einstellbar sein.

Zeig mir erst den Plan: welche Dateien, welche Einstellungen. Dann warte.



07 · Bibliothek

Die Skills, die Laurin selbst benutzt.

Nur Repos mit über 1.000 GitHub-Sternen oder offizielle von Anthropic und Shopify. Vor dem Installieren kurz reinschauen, was drin ist.

In der Desktop-App: Repo als ZIP laden, Skill-Ordner einzeln zippen (Ordner an die Wurzel, Beschreibung max. 200 Zeichen), dann Customize › Skills › „+“ › „Upload a skill“.

In Antigravity: `npx skills add <repo> -a antigravity -g -y --copy`

Laurins Favorit

Workflow

Laurin-Helfer

Pixel-Laurin in deinem Claude: beruhigt bei Frust, baut Briefs, führt durch Website- und Shopify-Abläufe.

Wann: Immer. Vor allem kurz bevor du Claude anschreiben willst.

[idefixart/saskia](https://github.com/idefixart/saskia)

github.com/idefixart/saskia

von Laurin

Laurins Favorit

Design

impeccable

Kritik- und Polish-Durchgang mit rund 60 automatischen Design-Checks.

Wann: Wenn die Seite steht, aber noch nicht gut aussieht. Befehle wie critique, polish, typeset, animate.

[pbakaus/impeccable](https://github.com/pbakaus/impeccable)

github.com/pbakaus/impeccable

82.268 Sterne

Laurins Favorit

Design

frontend-design

Plant Farben, Schrift und Abstände, bevor Code entsteht. Gegen generisches Design.

Wann: Ganz am Anfang einer neuen Seite oder Section.

[anthropics/skills](https://github.com/anthropics/skills)

github.com/anthropics/skills

181.729 Sterne

Design

ui-ux-pro-max

Nachschlagewerk mit 79 Stilen, 192 Paletten, Schriftpaaren und UX-Regeln.

Wann: Wenn du eine Richtung suchst und Ideen für Farben und Schriften brauchst.

[nextlevelbuilder/ui-ux-pro-max-skill](https://github.com/nextlevelbuilder/ui-ux-pro-max-skill)

github.com/nextlevelbuilder/ui-ux-pro-max-skill

136.800 Sterne

Design

taste-skill

Mehrere Skills gegen den KI-Template-Look, etwa high-end-visual-design und redesign-existing-projects.

Wann: Wenn eine Seite nach Baukasten aussieht oder eine alte Seite frisch werden soll.

[Leonxlnx/taste-skill](#)

github.com/Leonxlnx/taste-skill

98.345 Sterne

Design

hallmark

Gegen KI-Look, mit Audits. Übernimmt ein Design von einer URL oder einem Screenshot.

Wann: Wenn du eine Referenz hast und den Stil sauber übertragen willst.

[Nutlope/hallmark](#)

github.com/Nutlope/hallmark

30.270 Sterne

Design

MengTo Skills

no-ai-design-slop, Awwwards-taugliche Seiten und Scroll-Storytelling.

Wann: Für Landingpages, die richtig auffallen sollen.

[MengTo/Skills](#)

github.com/MengTo/Skills

6.810 Sterne

Animation

emil-design-eng

UI-Feinschliff und Bewegung, die sich hochwertig anfühlt. Dazu mobile-native.

Wann: Wenn Hover, Übergänge und Klicks sich noch billig anfühlen.

[emilkowalski/skills](#)

github.com/emilkowalski/skills

46.918 Sterne

Animation

GSAP-Skills

Offizielle Skills von GSAP für saubere Scroll- und Animationseffekte.

Wann: Wenn sich beim Scrollen etwas bewegen oder einblenden soll.

[greensock/gsap-skills](#)

[\(github.com/greensock/gsap-skills\)](#)

16.531 Sterne

Texte & Verkauf

marketingskills

copywriting, cro, launch, signup, marketing-psychology: Texte, die verkaufen.

Wann: Für Startseiten-Texte, Produktbeschreibungen und den Launch.

[coreyhaines31/marketingskills](#)

[\(github.com/coreyhaines31/marketingskills\)](#)

55.209 Sterne

Laurins Favorit

Qualität & SEO

web-quality-skills

Prüft Qualität, Core Web Vitals, Barrierefreiheit und SEO.

Wann: Vor dem Livegang und immer, wenn die Seite langsam wirkt.

[addyosmani/web-quality-skills](#)

[\(github.com/addyosmani/web-quality-skills\)](#)

2.956 Sterne

Qualität & SEO

web-design-guidelines

UI-Review mit Datei und Zeile, dazu React-Performance.

Wann: Wenn du wissen willst, was genau an der Oberfläche hakt.

[vercel-labs/agent-skills](#)

[\(github.com/vercel-labs/agent-skills\)](#)

32.556 Sterne

Laurins Favorit

Workflow

superpowers

Brainstorming, Pläne schreiben, systematisch debuggen und prüfen, bevor „fertig“ gesagt wird.

Wann: Bei jedem größeren Feature und jedem hartnäckigen Bug.

[obra/superpowers](#)

([github.com/obra/superpowers](#))

301.064 Sterne

Workflow

unlazy

Schreibt vorab Abnahme-Checks, damit nichts halb fertig zurückkommt.

Wann: Wenn Claude gern „fertig“ sagt, obwohl noch die Hälfte fehlt.

[Leonxlnx/unlazy](#)

([github.com/Leonxlnx/unlazy](#))

3.973 Sterne

Workflow

find-skills

Findet passende Skills und installiert sie.

Wann: Wenn du denkst: Dafür gibt es bestimmt schon einen Skill.

[vercel-labs/skills](#) ([github.com/vercel-labs/skills](#))

35.333 Sterne

Laurins Favorit

Shopify

Shopify AI Toolkit

Offiziell von Shopify: durchsucht die Doku, prüft GraphQL und Liquid, erledigt Store-Aufgaben per Shopify CLI.

Wann: Bei allem, was Themes, Liquid oder den Shop betrifft.

[Shopify/shopify-ai-toolkit](#)

([github.com/Shopify/shopify-ai-toolkit](#))

603 Sterne

Shopify

liquid-skills

Liquid-Grundlagen, Theme-Standards und Barrierefreiheit für Shopify-Themes.

Wann: Wenn du Sections und Snippets fürs Theme baust.

[Shopify/liquid-skills](#)

github.com/Shopify/liquid-skills

35 Sterne



Was Claude seit Kurzem kann, und wo du es findest.

Stand Oktober 2026, Desktop-App auf Windows. Menünamen in Anführungszeichen sind englisch, so wie in der App.

Projekte

„+ New Project“ › „Set project instructions“

Ein Projekt pro Website: Claude kennt Marke, Farben und Regeln, ohne dass du sie wiederholst. Jedes Projekt hat sein eigenes Memory.

Memory

Settings › Memory › Topics

Sieh nach, was Claude sich über dich gemerkt hat, und korrigier es.

Eigene Skills

Customize › Skills › „+“ › „Upload a skill“

Laurin-Helfer und Design-Skills direkt in der App, auf allen Plänen.

Claude Design

In jedem Chat und im Artifacts-Tab

Design-Entwürfe ausprobieren, als HTML oder ZIP exportieren und an Claude Code übergeben.

Research

„+“ › Research oder /deep-research

Konkurrenz und Markt gründlich recherchieren lassen. Bezahlte Pläne, frisst Limit.

Shopify-Connector

Customize › Connectors

Produkte, Lager und Rabatte per Chat. Nur mit dem Dev-Store testen.

Code-Tab

Desktop-App, bezahlter Plan

Claude Code ohne Terminal, mit Browser-Vorschau der laufenden Seite und Plan-Modus.

Modell und Effort

Menü neben dem Senden-Knopf

Sonnet und Low für Routine, Opus und High für harte Nüsse. Spart Limit.

Erweiterungen

Settings › Extensions

Hier läuft Pixel-Laurin. Eigene .mcpb-Dateien per Doppelklick installieren.

Verbrauch

Settings › Usage

Zeigt, wann dein Limit zurückgesetzt wird.



Symptom anklicken, Ursache lesen, Prompt kopieren.

Claude dreht sich im Kreis und macht dasselbe immer wieder kaputt

Ursache

Er rät Fixes, statt die Ursache zu suchen. Jeder Fehlversuch macht den Chat länger und frisst Limit.

Fix

Nach zwei Fehlversuchen stoppen. Erst die Ursache nennen lassen, dann genau ein Fix. Ist der Chat lang: Zusammenfassung, neuer Chat.

Stopp. Ändere noch nichts. Du hast das jetzt mehrmals versucht.
Lies die betroffenen Dateien noch einmal, nenn mir die
wahrscheinlichste Ursache und wie wir sie prüfen können.
Erst wenn die Ursache klar ist, schlägst du genau einen Fix vor.

Das Design sieht nach KI aus

Ursache

Ohne Referenzen und Vorgaben greift Claude zu Standards: lila Verläufe, drei gleiche Karten, überall dieselben Schatten.

Fix

Zwei, drei echte Referenzseiten geben und eine DESIGN.md anlegen. Dann gezielt einen Design-Skill wie impeccable drüberlaufen lassen.

Das sieht nach KI-Template aus. Vergleich die Seite mit diesen Referenzen: [Links].

Nenn mir die fünf größten Unterschiede bei Schrift, Abständen, Farben und Bildern.

Keine lila Verläufe, kein Hero mit drei gleichen Karten, keine Icons in runden Quadraten, kein reines Schwarz.

Erst die Liste, dann warten.

Claude vergisst Sachen, die wir längst geklärt haben

Ursache

Der Chat ist zu lang, oder die Info steht nur weit oben im Verlauf.

Fix

Was immer gilt, gehört in die Projekt-Anweisungen oder die DESIGN.md. Pro Thema ein neuer Chat, mit kurzer Zusammenfassung zum Start.

Fass alles zusammen, was wir in diesem Chat entschieden haben:

Ziel, Design-Entscheidungen, Dateien, offene Punkte.

Höchstens 15 Zeilen, so dass ich es in einen neuen Chat oder in die Projekt-Anweisungen kopieren kann.

Limit erreicht

Ursache

Lange Chats, große Anhänge, Research, hoher Effort und Opus fressen das 5-Stunden- und das Wochenlimit. App, Web und Claude Code teilen sich eins.

Fix

Unter Settings › Usage steht, wann es zurückgesetzt wird. Für Routine Sonnet oder niedrigeren Effort nehmen. Fragen bündeln, neuer Chat pro Thema.

Ich habe wenig Limit übrig. Antworte kurz, ohne Wiederholungen und ohne Zusammenfassung am Ende.
Frag nur nach, wenn du wirklich nicht weiterkommst.
Aufgabe: [...]

Claude ändert Dinge, die ich nicht wollte

Ursache

Die Aufgabe war offen formuliert, also hat er „mitgeholfen“. Oder er hat ohne Plan losgelegt.

Fix

Sag, was sich ändern darf und was tabu ist. Bei größeren Sachen erst den Plan zeigen lassen.

Ändere nur [Datei oder Bereich]. Alles andere bleibt genau so, wie es ist.
Zeig mir vorher, welche Dateien du anfassen willst und was du dort änderst.
Warte auf mein Okay.

Lokal geht's, live nicht

Ursache

Meist Cache, Groß- und Kleinschreibung in Dateinamen (Windows ist da tolerant, Server nicht), fehlende Dateien oder Pfade wie C:\...

Fix

Hart neu laden mit Strg+F5 oder im privaten Fenster testen. Dann Pfade prüfen lassen und in die Browser-Konsole schauen (F12).

Lokal funktioniert die Seite, live nicht: [Adresse, was genau kaputt ist].

Prüf alle Pfade auf Groß- und Kleinschreibung, auf absolute Windows-Pfade und auf Dateien, die nicht hochgeladen wurden. Sag mir, wonach ich in der Browser-Konsole (F12) schauen soll. Noch nichts ändern.

Meine Shopify-Änderung war sofort live

Ursache

Es wurde direkt im veröffentlichten Theme gearbeitet oder mit --live gepusht.

Fix

Nur im Dev-Store oder mit shopify theme dev arbeiten, Vorschau mit shopify theme push --unpublished. Live schaltest du selbst im Admin.

Ab jetzt gilt: Wir arbeiten nur mit shopify theme dev oder shopify theme push --unpublished.

Benutze nie --live, --publish oder --allow-live.

Bevor du einen Shopify-Befehl ausführst, zeig ihn mir und erkläre in einem Satz, was er tut.

Der Skill-Upload in der Desktop-App klappt nicht

Ursache

Meist ist die Beschreibung länger als 200 Zeichen, oder der Ordner liegt nicht an der ZIP-Wurzel oder heißt anders als der Skill.

Fix

Settings › Capabilities › „Code execution and file creation“ muss an sein. Den Ordner selbst zippen, nicht seinen Inhalt. Beschreibung notfalls kürzen lassen.

```
Hier ist die SKILL.md eines Skills: [einfügen]
Kürze die description auf höchstens 200 Zeichen und behalte die
wichtigsten Auslöse-Wörter.
Prüf, dass name nur Kleinbuchstaben und Bindestriche hat und
genauso heißt wie der Ordner.
```

Claude ist in Antigravity nicht auswählbar

Ursache

Opus 5.5 und Sonnet 5.5 gibt es dort nur mit Google AI Pro (nicht im Test), Ultra oder Enterprise. Die 4.6-Modelle fliegen am 02.11.2026 raus.

Fix

Google-Abo prüfen. Oder im Extensions-Panel die offizielle Erweiterung „Claude Code“ installieren, die läuft über dein Claude-Abo.

```
Du läufst als Claude Code in Antigravity. Lies zuerst die Regeln
in .agents/rules/ und die DESIGN.md im Projekt.
Fass in drei Sätzen zusammen, was davon für diese Aufgabe
wichtig ist.
Dann zeig mir einen Plan für: [Aufgabe]. Noch nichts ändern.
```

Antigravity ignoriert meine Regeln

Ursache

Dateien in `.agents/rules/` brauchen oben ein Frontmatter mit `trigger`, sonst werden sie still übergangen.

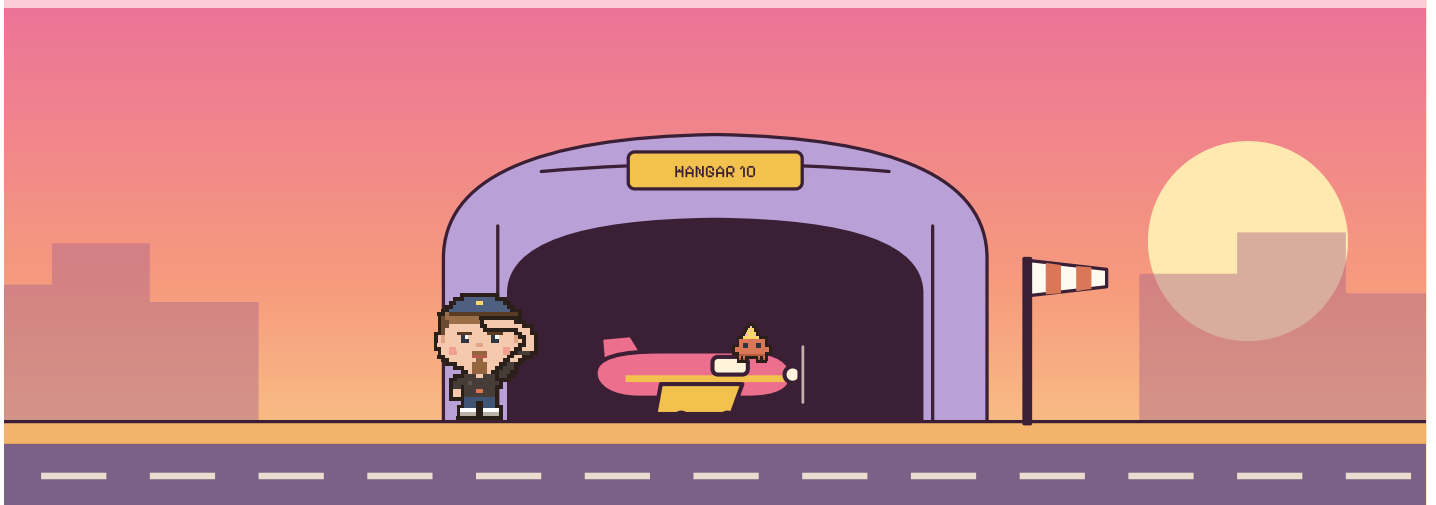
Fix

Ganz oben in die Datei: `trigger: always_on`, eingerahmt von zwei Zeilen mit `---`. Jede Datei unter 24 KB halten.

```
Prüf alle Dateien in .agents/rules/: Hat jede oben ein
Frontmatter mit trigger (always_on, model_decision, glob oder
manual)?
```

```
Ist eine größer als 24 KB?
```

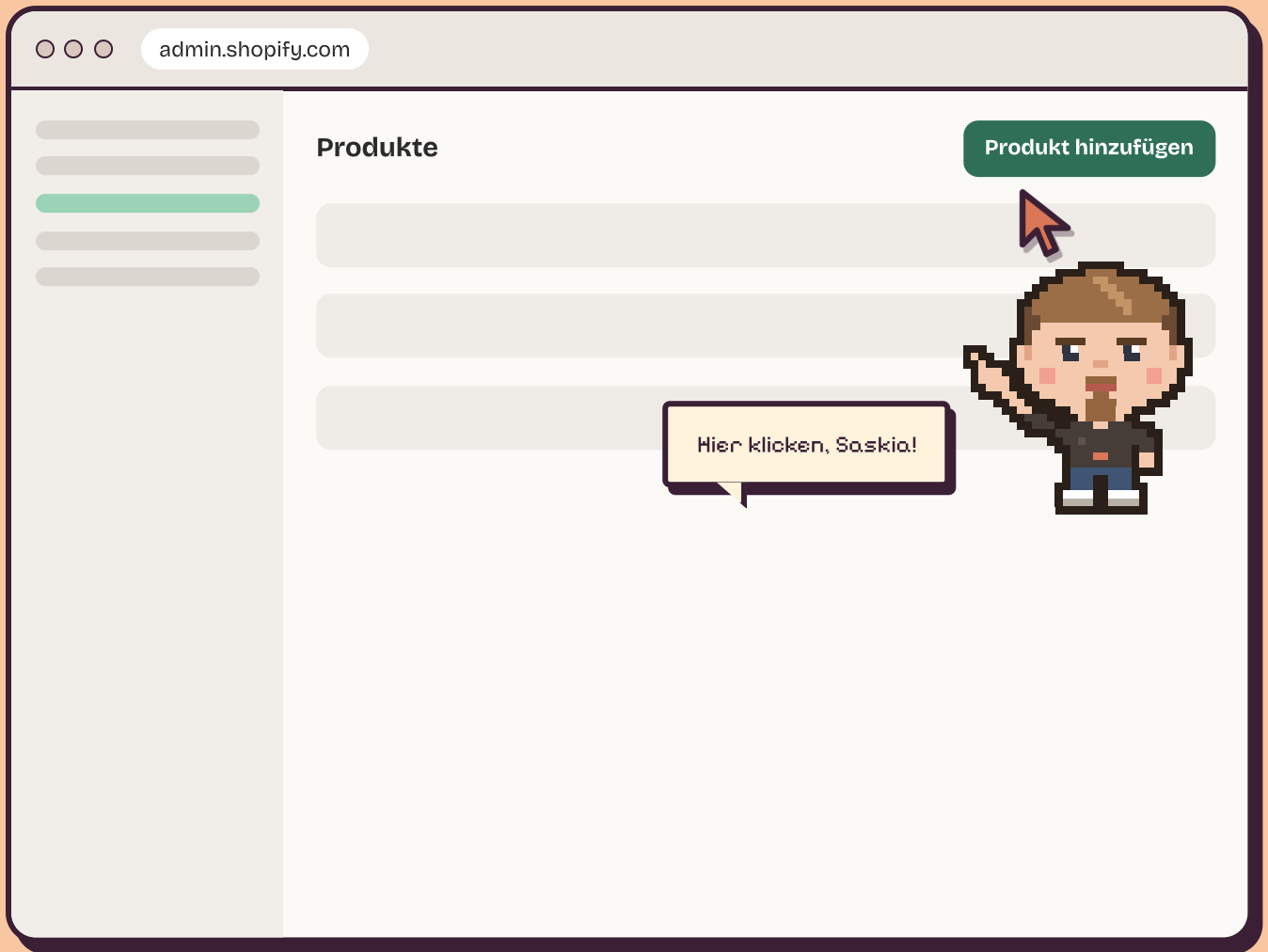
```
Liste die Probleme auf, ändere noch nichts.
```



10 · Buddy-Hangar

Pixel-Laurin auf deinem Bildschirm.

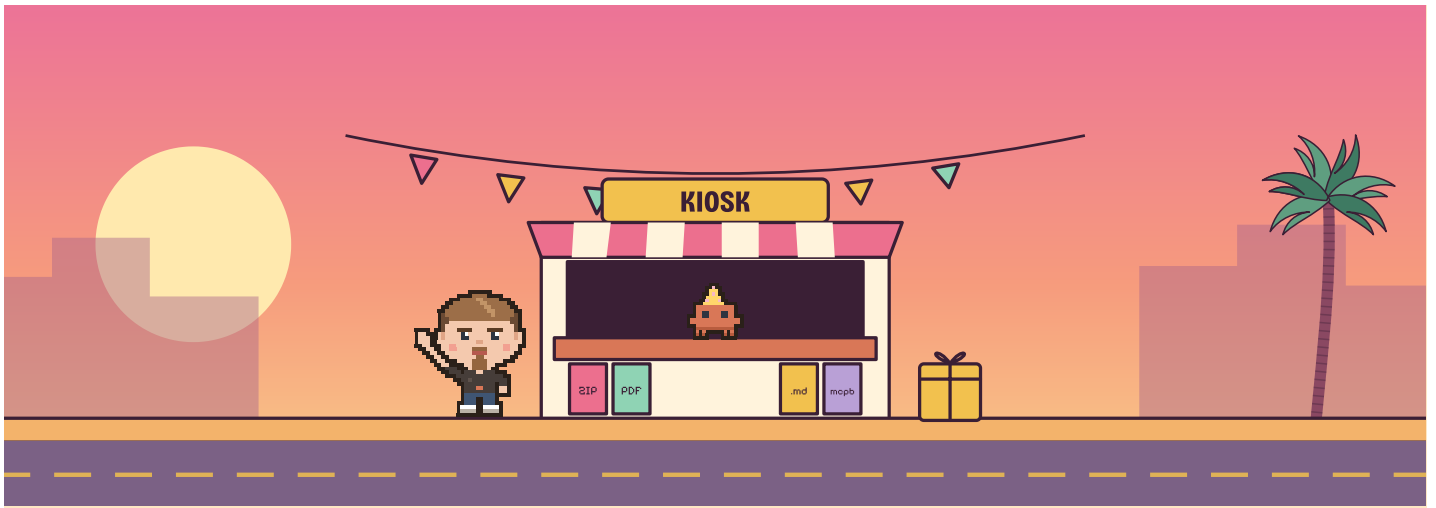
Er schwebt über deinen Bildschirm und zeigt mit einem Pfeil, wo du klicken musst. Claude schaut sich deinen Bildschirm an, zeigt dir einen Schritt, du klickst und sagst „weiter“. Dann kommt der nächste.



Installieren (Windows, Claude-Desktop-App)

- 1 [laurin-buddy.mcpb](#) herunterladen.
- 2 Doppelklick auf die Datei, Claude Desktop installiert sie. Alternativ: Settings › Extensions › Advanced settings › Extension Developer › „Install Extension...“
- 3 Wenn Claude fragt, die Werkzeuge erlauben.
- 4 Ausprobieren: „Zeig mir, wo ich in Shopify ein Produkt anlege.“

Er klickt nie selbst. Er zeigt nur. Einen Screenshot gibt es nur, wenn du ihn darum bittest.



11 · Kiosk

Alles zum Runterladen.



Laurin-Helfer

Pixel-Laurin als Skill in deinem Claude.

Desktop-App: Settings › Capabilities › „Code execution and file creation“ an, dann Customize › Skills › „+“ › „+ Create skill“ › „Upload a skill“.

Antigravity: entpacken, Ordner nach `C:\Users\
<Name>\.gemini\config\skills\` kopieren.

Download: idefixart.github.io/saskia/#starterkit

Laurin-Buddy

Zeigt dir auf dem Bildschirm, wo du klicken musst.

Doppelklick, Claude Desktop installiert ihn.

Download: idefixart.github.io/saskia/#starterkit

Projekt-Anweisungen

Laurins Regeln für ein Projekt „Websites & Shops“.

„+ New Project“ › „Set project instructions“, Text reinkopieren.

Download: idefixart.github.io/saskia/#starterkit

Antigravity-Regeln

Dieselben Regeln, zum Bauen.

Als `.agents/rules/laurin.md` ins Projekt oder global nach `C:\Users\
<Name>\.gemini\config\rules\laurin.md`.

Download: idefixart.github.io/saskia/#starterkit

Brief-Vorlage

Zum Ausfüllen vor jedem neuen Projekt.

Ausfüllen und in einen neuen
Chat kopieren.

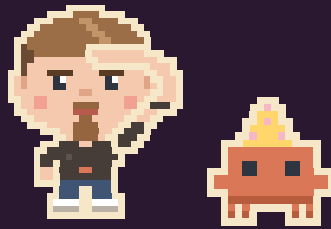
Download: idefixart.github.io/saskia/#starterkit

Alles als PDF

Die ganze Seite zum Ausdrucken.

Öffnen, lesen, an die Wand hängen.

Download: idefixart.github.io/saskia/#starterkit



Bis später in Claude-Town.

Ein Fan-Geschenk von Laurin. Nicht von Anthropic.

 **Nochmal auspacken**

Gebaut am 17.10.2026, natürlich mit Claude.