

GhibliKitchen – Wochenpläne & Kochbuch-Layouts

GhibliKitchen ist ein Koch- und Ernährungs-Assistent für strukturierte Wochenpläne, Einkaufslisten und bebilderte Kochbuch-Layouts – mit Fokus auf **CN/JP/KR-Küche**, mild gewürzt und angepasst an **Diabetes (früh)**, **Gastritis (mild)** und **Schwangerschaft**.

Das Projekt verbindet klar definierte **Gesundheits-Leitplanken**, ein **druckfähiges UI (PDF/HTML)** und eine **Datenstruktur, die von einem GPT („GhibliKitchen“) erzeugt/validiert werden kann**.

Inhalt

- [Ziele](#)
 - [Gesundheits-Leitplanken \(streng\)](#)
 - [Küchenfokus & Stil](#)
 - [Technik-Stack & Architektur](#)
 - [Layout](#)
 - [.jsx-Datei & UI-Layout](#)
 - [Datenmodell \(Rezepte & Einkaufsliste\)](#)
 - [Verknüpfung zu GPT „GhibliKitchen“](#)
 - [Entwicklung & Nutzung](#)
 - [Qualitätsregeln & Validierung](#)
 - [Beitragen](#)
 - [Rechtliches & Haftung](#)
-

Ziele

- Wöchentlich **21 Rezepte** (7 Tage × Frühstück/Mittag/Abend) für 2 Personen.
 - **Druckfertige** Kochbuch-Seiten (A4 quer, 1 Rezept = 1 Seite) + Einkaufslisten-Export.
 - **Milde, magenfreundliche** Zubereitung; **salzarm, ohne Schärfe**.
 - **Automatisierte Einkaufsliste** aus allen Rezepten (konsistente Einheiten/Benennungen).
-

Gesundheits-Leitplanken (streng)

- **Diabetes (frühes Stadium; Metformin 3× täglich)**
Ziel pro Rezept (2 P.): **60–90 g KH gesamt** (≈30–45 g p. P.), ballaststoffbetont; **Protein 20–40 g p. P.**
Metformin-Hinweis: reine **Erinnerung „mit der Mahlzeit“**, **keine** Med-Beratung.
- **Gastritis (mild)**
Meiden: Chili/Schärfe, stark Fettiges, Zitrone/Essig/zu saure Tomaten, Alkohol, Kaffee, Kohlensäure.
Methoden: **Dämpfen, Sieden, Schmoren, sanftes Dünsten**; Zwiebel/Knoblauch **sparsam & gut gegart**; milde Brühen; **wenig Öl**; **warme Speisen**.
- **Schwangerschaft**
Kein Rohes, alles **durchgaren** (Eier vollständig gestockt), **quecksilberarme Fische** (Lachs/

Kabeljau/Seelachs/Wolfsbarsch), Hygiene; **Sojasauce natriumarm, Jod** (Wakame/Kombu) sparsam.

Küchenfokus & Stil

- **Primär:** Chinesisch (CN), Japanisch (JP), Koreanisch (KR).
 - **Sekundär (max. 1 Gericht/Woche):** leichte IT-Gerichte (z. B. Minestrone, mildes Vollkorn-Risotto, Zoodles).
 - **Stil & Zubereitung:** mild, salzarm; wenn braten, dann wenig Öl; Säure mildern (länger köcheln, Karotte/Brühe).
 - **Titel-Format je Gericht:** *Deutsch + Originalname + Schriftzeichen* (z. B. „Miso-Suppe (味噌汁)“).
 - **Inspirationsquellen (nur als Inspiration):** Just One Cookbook, maff.go.jp, market/k_ryouri, My Korean Kitchen, Seon Kyoung Longest, Mom's Korean Recipes, Made With Lau, The Woks of Life, Omnivores Cookbook, Redhouse Spice, Lee Kum Kee.
-

Technik-Stack & Architektur

- **React (Single-File JSX)** – rendert Tabs *Kochbuch* (A4 quer) & *Einkaufsliste* (A4 hoch).
 - **Client-Export** via `html2pdf.js@0.10.1` (CDN-geladen).
 - **State & Storage:** Bilder-Uploads via **FileReader**, Persistenz in `localStorage`.
 - **Design Tokens:** Nur HEX/RGBA, keine modernen CSS-Mix-Funktionen.
-

Layout

- **Grid:** 12 Spalten; Panel links **span 4**, Rezept rechts **span 8**
 - **Cards:** `borderRadius: 18`, `Padding 20/22`, `boxShadow: COLORS.btnShadow`
-

.jsx-Datei & UI-Layout

Beispiel-Datei: `Woche-2-2025-10-06.jsx`

```
export const meta = { title: "Woche 2", startDate: "2025-10-06", id:
"woche-2-2025-10-06" };
const FILE_BASE = "Woche 2 2025-10-06";
// UI-Titel:
// - Kochbuch-Tab: "GhibliKitchen - Woche 2"
// - Einkaufsliste: "GhibliKitchen - Einkaufsliste - Woche 2"
```

- **TopBar-Buttons (nur drei):** „PDF erzeugen“, „HTML exportieren“, „Drucken“
- **Nach PDF-Erzeugung:** Download-Link unter dem jeweiligen Tab-Inhalt
- **DALL-E-Prompts:** werden nicht gerendert, nur als Strings im Code vorgehalten
- **Cover & Rezepte:** Upload möglich, Speicherung in `localStorage`

Seitenstruktur

- **A4 quer (Kochbuch):** 1 Rezept = 1 Seite; Panel links $\leq 1/3$, Rezept rechts $\geq 2/3$
- **A4 hoch (Einkaufsliste)**

Datenmodell (Rezepte & Einkaufsliste)

- **Genau 7 Tage \times 3 Meals = 21 Rezepte**
- **IDs:** `mo|di|mi|do|fr|sa|so + -f|-m|-a`

Objekt je Rezept

```
{
  id: string, // z. B. "mo-f"
  title: string, // "Deutsch + Original + 字"
  target: { carbs: string, proteinPerPerson?: string }, // KH gesamt (2 P.),
optional Protein p. P.
  ingredients: Array<{ name: string, qty: number, unit: "g"|"ml"|"Stk",
notes?: string }>, //  $\geq 5$  Einträge
  steps: string[], //  $\geq 3$  Schritte, nummeriert im UI
  checks: string, // "Gastritis ✓/! ... · Diabetes ✓/! -  $\approx$ XX g KH ·
Schwangerschaft ✓/! ..."
  swaps: string, // sinnvolle Alternativen
  side: string, // milde Beilage/Drink
  remind: boolean // Metformin-Reminder: Frühstück/Abend true, Mittag false
}
```

Einkaufsliste (`LIST_SUMMARY`)

Automatisch aggregiert & **gruppiert**:

- Protein/Fisch/Tofu
- Gemüse/Pilze
- Reis/Nudeln/Sättigung
- Algen/Brühen/Würze

Mengen werden aus allen Rezepten **arithmetisch konsistent aufsummiert**
(z. B. „Reis (roh)“, „Sojasauce natriumarm“)

Verknüpfung zu GPT „GhibliKitchen“

- **Rolle:** Ein spezialisiertes GPT erzeugt/validiert Wochenpläne gemäß obiger Leitplanken

Aufgaben des GPT

- Rezepte generieren (CN/JP/KR-dominant, max. 1 IT-Gericht/Woche)
- **KH-Ziel 60–90 g** (2 P.) pro Rezept sicherstellen; Protein 20–40 g p. P. optional angegeben

- **Sicherheit:** keine rohen Zutaten; Fisch/Ei vollständig gegart; Jod sparsam
- **Metformin-Logik:** nur Frühstück/Abend `remind: true`, Mittag `false`
- **Output-Konformität:** strukturierte Objekte, korrekte Einheiten, ≥ 5 Zutaten, ≥ 3 Schritte, Checks/Swaps/Side vorhanden

Nicht-Ziel: medizinische Beratung – ausschließlich **UI-Reminder** „mit der Mahlzeit“

Entwicklung & Nutzung

Projekt lokal starten

1. React-Projekt vorbereiten (Vite, CRA o. ä.)
2. Datei z. B. `src/Woche-2-2025-10-06.jsx` hinzufügen
3. Komponente in der App mounten:

```
import Woche2 from "../Woche-2-2025-10-06.jsx";
export default function App(){ return <Woche2 />; }
```

1. Starten: `npm run dev` (oder projektabhängiges Skript)

Export & Druck

- **PDF:** „PDF erzeugen“ (Kochbuch = A4 quer, Einkaufsliste = A4 hoch)
- **HTML:** „HTML exportieren“ (Standalone HTML mit eingebetteten Styles)
- **Drucken:** System-Dialog; Upload-Kontrollen sind per `.print:hidden` ausgeblendet

Bilder

- Cover & Rezeptbilder per Upload (FileReader) → `localStorage` persistiert
- Jeder Slot kann zurückgesetzt werden („Bild löschen“)

Qualitätsregeln & Validierung

Automatische UI-Tests (beim Mount in der Konsole):

- 21 eindeutige IDs
- Meal-Mapping korrekt
- Metformin-Logik korrekt
- Farben ohne verbotene CSS-Funktionen
- `FILE_BASE` Regex: `^Woche 2 \d{4}-\d{2}-\d{2}$`
- `buildPrompt("A", "B") === "A\nB"`

Bei Erfolg:

```
[GhibliKitchen] All tests passed (JSX).
```

Beitragen

- **Neue Wochen:** neue `.jsx` mit entsprechender `meta.startDate` und `FILE_BASE`
 - **Rezepte erweitern:** Leitplanken beachten; mind. 5 Zutaten, 3 Schritte, Checks/Swaps/Side pflegen
 - **i18n:** Umlaute & CJK-Zeichen korrekt (中/日/韓; Kanji/Hanzi/Hangul)
 - **Barrierefreiheit:** Alternativtexte für Bilder
-

Rechtliches & Haftung

- **Kein medizinischer Rat.** Inhalte dienen der Planung & Dokumentation; individuelle Bedürfnisse können abweichen
 - Marken- & Urheberrechte der genannten Inspirationsquellen verbleiben bei den jeweiligen Inhabern
 - Beiträge müssen frei von urheberrechtlich geschütztem Material Dritter sein (ohne Lizenz)
-

Kurzfassung

GhibliKitchen bietet eine robuste, GPT-gestützte Pipeline für **gesunde, milde Wochenpläne** mit **druckfertigem Layout** und **automatischer Einkaufsliste** – alles in **einer React-Datei** pro Woche