

Vibe Coding Challenge:

CO2 Browser-Game

Aufgabe

Konzipiere und entwickle ein Browser-Game (HTML + Three.js), das spielerisch vermittelt, wie viele Bäume ein einzelner Mensch in Deutschland pflanzen müsste, um seinen eigenen CO2 Abdruck zu neutralisieren. Das Spiels soll interaktiv vermitteln, wie viel Tonnen CO2-Emission ein einzelner Mensch im Durchschnitt verursacht und was man dagegen tun kann.

Empfohlene Vorgehensweise:

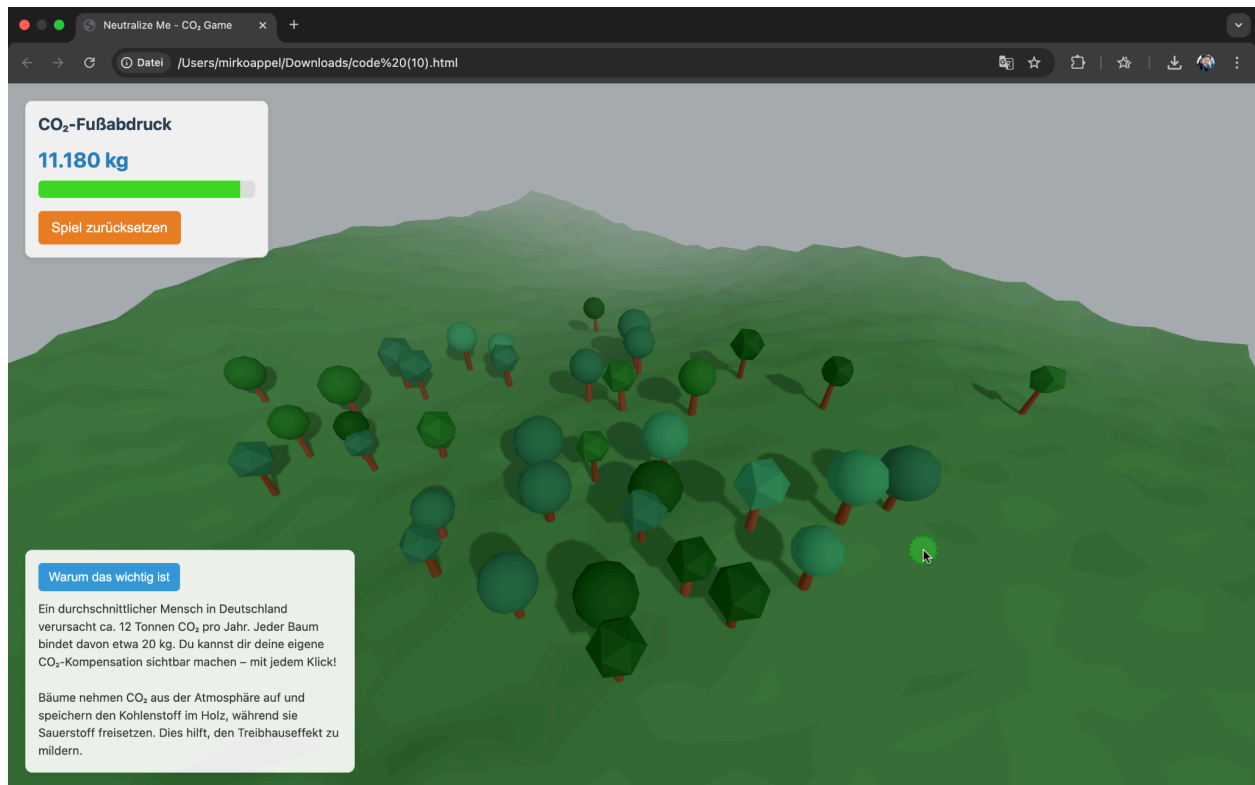
1. Konkretisiere deine Spielidee und erstelle dann ein detailliertes Briefing für deinen Programmierer (Gemini), damit er das Spiel für dich programmieren kann. Tipp: KI ist hier Freund und Helfer!
2. Öffne Google AI Studio und wähle das KI-Modell Gemini 2.5 Pro Preview aus. Gemini (dein Entwickler) ist nun bereit und wartet auf dein Briefing.
3. Teile das Briefing mit Gemini, damit die KI das Spiel für dich entwickeln kann.
4. Laden den von Gemini entwickelten Code herunter und öffne die heruntergeladene HTML-Datei in deinem Browser (Doppelklick), um das Spiel zu testen.
5. Passe das Spiel im Dialog mit Gemini weiter an (optional)

Prompt zur Entwicklung des Briefings (Anweisungen für Gemini)

— nach eingehender Recherche und Brainstorming mit ChatGPT —

Okay, lasst uns da draußen ein Spiel machen. Der Spieler soll so lange Bäume pflanzen, bis sein eigener CO₂-Fußabdruck neutralisiert ist. Das Ganze wird mit Tree.js umgesetzt in einer 3D-Welt mit einer Kamerasteuerung. Mit einem Klick pflanzt man einen Baum. Der entsteht dann auch in der Szene. Und dann gibt es irgendwo in einem Barometer, was anzeigt, wie CO₂-neutral man schon ist. Und irgendwo gibt es auch eine Infobox, die man aufklappen kann, wo das auch erklärt wird. Vielleicht entsteht ein Popup oder so. Mache daraus ein richtig gutes Briefing, das ich benutzen kann für den Spieleentwickler, der das Ganze umsetzen soll.

Musterlösung



KI-Tools:

1. ChatGPT 4o ← Recherche, Konzeption, Visualisierung und Entwicklung des Briefings für Gemini zur Entwicklung des Spiels
2. Google AI Studio mit Gemini 2.5 Pro Preview ← Entwicklung des Codes für das Spiel

Prompt zur Entwicklung des Spiels (Anweisungen für Gemini):

Projektbriefing: CO₂-Game "Neutralize Me"

Ein interaktives Baum-Pflanz-Spiel zur CO₂-Kompensation

Technologie: Three.js (WebGL-basiert)

Ziel des Spiels

Der Spieler erfährt spielerisch, wie viele Bäume gepflanzt werden müssen, um den durchschnittlichen CO₂-Fußabdruck eines Menschen in Deutschland zu neutralisieren (~12 Tonnen pro Jahr). Jeder gepflanzte Baum senkt diesen Wert. Ziel: auf 0 kg CO₂ zu kommen.

Core Gameplay

- + Spielprinzip: Der Spieler klickt auf den Boden → an dieser Stelle wird ein 3D-Baum erzeugt (visuell animiert, idealerweise mit sanftem Wachstumseffekt). Jeder Baum reduziert den CO₂-Zähler um 20 kg CO₂ (optimistischer Mittelwert).
- + Ziel: CO₂-Barometer auf 0 bringen. Standardwert beim Start: 12.000 kg CO₂ (12 Tonnen).

Interaktionen

Steuerung

- + Kamera: Frei beweglich (OrbitControls), Zoom, Rotation. Spieler kann sich umsehen, aber nicht durch die Welt fliegen.
- + Baum-Pflanzung: Linksklick auf Terrain → Baum wächst an dieser Stelle (ggf. Raycasting).
- + Optional: Cursor zeigt Baum-Vorschau vor dem Pflanzen.

Visuelle Elemente

Die Spielwelt

- + Eine einfache, stylisierte 3D-Fläche (Low Poly / Flat Color) als Terrain.
- + Leichte Hügelstruktur oder Insel, damit es nicht zu steril wirkt.
- + Optional: Lichtveränderung bei CO₂-Fortschritt (z. B. Smog → klarer Himmel).

Die Bäume

- + Stylisierte Modelle (z. B. einfache Kugel-Krone + Stamm).
- + Variationen (zufällige Größe, Farbe, Typ).
- + Animiertes Wachsen beim Pflanzen.

UI-Elemente

1. CO₂-Barometer (HUD oben oder seitlich)

- + Zeigt: Verbleibender CO₂-Fußabdruck: XXXX kg
- + Bei jedem Baum wird der Wert reduziert.
- + Fortschrittsbalken (0 % = neutralisiert).

2. Infobox / Popup (klappbar)

- + Enthält eine Erklärung, warum Bäume CO₂ kompensieren.
- + Beispieltext: „Ein durchschnittlicher Mensch in Deutschland verursacht ca. 12 Tonnen CO₂ pro Jahr. Jeder Baum bindet davon etwa 20 kg. Du kannst dir deine eigene CO₂-Kompensation sichtbar machen – mit jedem Klick!“
- + Button: “Warum das wichtig ist” → öffnet/klappt Text auf.

3. Reset-Button

- + Setzt Spielfortschritt zurück (Zähler + Welt).

Lerninhalte (als Teil der UI oder Popups)

- + Durchschnittlicher CO₂-Fußabdruck
- + Bindung pro Baum (realistische Werte)
- + Optional: Weitere Emissionsquellen (Auto, Flug, Kleidung – mit Buttons/Piktogrammen als Bonus-Content)

Technische Anforderungen

- + Technologie: Three.js, HTML/CSS/JS
- + Responsiv: Mindestens Desktop, optional mobilfähig
- + Kein Backend notwendig (nur Client-seitige Logik)
- + Assetverwaltung: Lokale Models oder CDN-basiert

Stretch Goals (Optional)

- + Tiere erscheinen, wenn CO₂-Wert sinkt (Gamification).
- + Mini-Statistik: "Du hast so viel CO₂ gebunden wie ... X Burger, Y Flugmeilen, etc."
- + Screenshot-Funktion: Spieler kann seinen neutralisierten Garten teilen.

Erfolgskriterium

- + Spaßiges, meditatives Gameplay mit realem Lerneffekt.
- + Gute Performance im Browser.
- + Funktionierender Loop: CO₂ reduzieren → Bäume pflanzen → Fortschritt sehen → Aha-Erlebnis.

Erstelle alles in einer einzigen HTML-Datei