On utilisont cette hero section design et appliquer tout le siteweb entier :

"use client";

import React, { useEffect, useRef } from "react";

const colors = {

50: "#f8f7f5",

100: "#e6e1d7",

200: "#c8b4a0",

300: "#a89080",

400: "#8a7060",

500: "#6b5545",

600: "#544237",

700: "#3c4237",

800: "#2a2e26",

900: "#1a1d18",

};

export function Component() {

const gradientRef = useRef<HTMLDivElement>(null);

useEffect(() => {

// Animate words

const words = document.querySelectorAll<HTMLElement>(".word");

words.forEach((word) => {

const delay = parseInt(word.getAttribute("data-delay") || "0", 10);

setTimeout(() => {

word.style.animation = "word-appear 0.8s ease-out forwards";

}, delay);

});

// Mouse gradient

const gradient = gradientRef.current;

function onMouseMove(e: MouseEvent) {

if (gradient) {

gradient.style.left = e.clientX - 192 + "px";

gradient.style.top = e.clientY - 192 + "px";

gradient.style.opacity = "1";

}

}

function onMouseLeave() {

if (gradient) gradient.style.opacity = "0";

}

document.addEventListener("mousemove", onMouseMove);

document.addEventListener("mouseleave", onMouseLeave);

// Word hover effects

words.forEach((word) => {

word.addEventListener("mouseenter", () => {

word.style.textShadow = "0 0 20px rgba(200, 180, 160, 0.5)";

});

word.addEventListener("mouseleave", () => {

word.style.textShadow = "none";

});

});

// Click ripple effect

function onClick(e: MouseEvent) {

const ripple = document.createElement("div");

ripple.style.position = "fixed";

ripple.style.left = e.clientX + "px";

ripple.style.top = e.clientY + "px";

ripple.style.width = "4px";

ripple.style.height = "4px";

ripple.style.background = "rgba(200, 180, 160, 0.6)";

ripple.style.borderRadius = "50%";

ripple.style.transform = "translate(-50%, -50%)";

ripple.style.pointerEvents = "none";

ripple.style.animation = "pulse-glow 1s ease-out forwards";

document.body.appendChild(ripple);

setTimeout(() => ripple.remove(), 1000);

}

document.addEventListener("click", onClick);

// Floating elements on scroll

let scrolled = false;

function onScroll() {

if (!scrolled) {

scrolled = true;

document.querySelectorAll<HTMLElement>(".floating-element").forEach((el, index) => {

setTimeout(() => {

el.style.animationPlayState = "running";

}, index \* 200);

});

}

}

window.addEventListener("scroll", onScroll);

return () => {

document.removeEventListener("mousemove", onMouseMove);

document.removeEventListener("mouseleave", onMouseLeave);

document.removeEventListener("click", onClick);

window.removeEventListener("scroll", onScroll);

};

}, []);

return (

<div

className="min-h-screen bg-gradient-to-br from-[#1a1d18] via-black to-[#2a2e26] text-[#e6e1d7] font-primary overflow-hidden relative w-full"

>

<svg className="absolute inset-0 w-full h-full" xmlns="http://www.w3.org/2000/svg">

<defs>

<pattern id="grid" width="60" height="60" patternUnits="userSpaceOnUse">

<path

d="M 60 0 L 0 0 0 60"

fill="none"

stroke="rgba(200,180,160,0.08)"

strokeWidth="0.5"

/>

</pattern>

</defs>

<rect width="100%" height="100%" fill="url(#grid)" />

<line x1="0" y1="20%" x2="100%" y2="20%" className="grid-line" style={{ animationDelay: "0.5s" }} />

<line x1="0" y1="80%" x2="100%" y2="80%" className="grid-line" style={{ animationDelay: "1s" }} />

<line x1="20%" y1="0" x2="20%" y2="100%" className="grid-line" style={{ animationDelay: "1.5s" }} />

<line x1="80%" y1="0" x2="80%" y2="100%" className="grid-line" style={{ animationDelay: "2s" }} />

<line

x1="50%"

y1="0"

x2="50%"

y2="100%"

className="grid-line"

style={{ animationDelay: "2.5s", opacity: 0.05 }}

/>

<line

x1="0"

y1="50%"

x2="100%"

y2="50%"

className="grid-line"

style={{ animationDelay: "3s", opacity: 0.05 }}

/>

<circle cx="20%" cy="20%" r="2" className="detail-dot" style={{ animationDelay: "3s" }} />

<circle cx="80%" cy="20%" r="2" className="detail-dot" style={{ animationDelay: "3.2s" }} />

<circle cx="20%" cy="80%" r="2" className="detail-dot" style={{ animationDelay: "3.4s" }} />

<circle cx="80%" cy="80%" r="2" className="detail-dot" style={{ animationDelay: "3.6s" }} />

<circle cx="50%" cy="50%" r="1.5" className="detail-dot" style={{ animationDelay: "4s" }} />

</svg>

{/\* Corner elements \*/}

<div className="corner-element top-8 left-8" style={{ animationDelay: "4s" }}>

<div

className="absolute top-0 left-0 w-2 h-2 opacity-30"

style={{ background: colors[200] }}

></div>

</div>

<div className="corner-element top-8 right-8" style={{ animationDelay: "4.2s" }}>

<div

className="absolute top-0 right-0 w-2 h-2 opacity-30"

style={{ background: colors[200] }}

></div>

</div>

<div className="corner-element bottom-8 left-8" style={{ animationDelay: "4.4s" }}>

<div

className="absolute bottom-0 left-0 w-2 h-2 opacity-30"

style={{ background: colors[200] }}

></div>

</div>

<div className="corner-element bottom-8 right-8" style={{ animationDelay: "4.6s" }}>

<div

className="absolute bottom-0 right-0 w-2 h-2 opacity-30"

style={{ background: colors[200] }}

></div>

</div>

{/\* Floating elements \*/}

<div className="floating-element" style={{ top: "25%", left: "15%", animationDelay: "5s" }}></div>

<div className="floating-element" style={{ top: "60%", left: "85%", animationDelay: "5.5s" }}></div>

<div className="floating-element" style={{ top: "40%", left: "10%", animationDelay: "6s" }}></div>

<div className="floating-element" style={{ top: "75%", left: "90%", animationDelay: "6.5s" }}></div>

<div className="relative z-10 min-h-screen flex flex-col justify-between items-center px-8 py-12 md:px-16 md:py-20">

{/\* Top tagline \*/}

<div className="text-center">

<h2

className="text-xs md:text-sm font-mono font-light uppercase tracking-[0.2em] opacity-80"

style={{ color: colors[200] }}

>

<span className="word" data-delay="0">

Welcome

</span>

<span className="word" data-delay="200">

to

</span>

<span className="word" data-delay="400">

<b>StackPilot</b>

</span>

<span className="word" data-delay="600">

—

</span>

<span className="word" data-delay="800">

Powering

</span>

<span className="word" data-delay="1000">

your

</span>

<span className="word" data-delay="1200">

digital

</span>

<span className="word" data-delay="1400">

transformation.

</span>

</h2>

<div

className="mt-4 w-16 h-px opacity-30"

style={{

background: `linear-gradient(to right, transparent, ${colors[200]}, transparent)`,

}}

></div>

</div>

{/\* Main headline \*/}

<div className="text-center max-w-5xl mx-auto">

<h1

className="text-3xl md:text-5xl lg:text-6xl font-extralight leading-tight tracking-tight text-decoration"

style={{ color: colors[50] }}

>

<div className="mb-4 md:mb-6">

<span className="word" data-delay="1600">

Supercharge

</span>

<span className="word" data-delay="1750">

your

</span>

<span className="word" data-delay="1900">

productivity

</span>

<span className="word" data-delay="2050">

with

</span>

<span className="word" data-delay="2200">

AI-driven

</span>

<span className="word" data-delay="2350">

automation.

</span>

</div>

<div

className="text-2xl md:text-3xl lg:text-4xl font-thin leading-relaxed"

style={{ color: colors[200] }}

>

<span className="word" data-delay="2600">

Integrate,

</span>

<span className="word" data-delay="2750">

orchestrate,

</span>

<span className="word" data-delay="2900">

and

</span>

<span className="word" data-delay="3050">

scale

</span>

<span className="word" data-delay="3200">

your

</span>

<span className="word" data-delay="3350">

business

</span>

<span className="word" data-delay="3500">

— all

</span>

<span className="word" data-delay="3650">

in

</span>

<span className="word" data-delay="3800">

one

</span>

<span className="word" data-delay="3950">

secure

</span>

<span className="word" data-delay="4100">

platform.

</span>

</div>

</h1>

<div

className="absolute -left-8 top-1/2 w-4 h-px opacity-20"

style={{

background: colors[200],

animation: "word-appear 1s ease-out forwards",

animationDelay: "3.5s",

}}

></div>

<div

className="absolute -right-8 top-1/2 w-4 h-px opacity-20"

style={{

background: colors[200],

animation: "word-appear 1s ease-out forwards",

animationDelay: "3.7s",

}}

></div>

</div>

{/\* Bottom tagline \*/}

<div className="text-center">

<div

className="mb-4 w-16 h-px opacity-30"

style={{

background: `linear-gradient(to right, transparent, ${colors[200]}, transparent)`,

}}

></div>

<h2

className="text-xs md:text-sm font-mono font-light uppercase tracking-[0.2em] opacity-80"

style={{ color: colors[200] }}

>

<span className="word" data-delay="4400">

Real-time

</span>

<span className="word" data-delay="4550">

analytics,

</span>

<span className="word" data-delay="4700">

seamless

</span>

<span className="word" data-delay="4850">

integrations,

</span>

<span className="word" data-delay="5000">

enterprise-grade

</span>

<span className="word" data-delay="5150">

security.

</span>

</h2>

<div

className="mt-6 flex justify-center space-x-4 opacity-0"

style={{

animation: "word-appear 1s ease-out forwards",

animationDelay: "4.5s",

}}

>

<div

className="w-1 h-1 rounded-full opacity-40"

style={{ background: colors[200] }}

></div>

<div

className="w-1 h-1 rounded-full opacity-60"

style={{ background: colors[200] }}

></div>

<div

className="w-1 h-1 rounded-full opacity-40"

style={{ background: colors[200] }}

></div>

</div>

</div>

</div>

<div

id="mouse-gradient"

ref={gradientRef}

className="fixed pointer-events-none w-96 h-96 rounded-full blur-3xl transition-all duration-500 ease-out opacity-0"

style={{

background: `radial-gradient(circle, ${colors[500]}0D 0%, transparent 100%)`,

}}

></div>

</div>

);

}

@import "tailwindcss";

@import "tw-animate-css";

@keyframes word-appear {

0% {

opacity: 0;

transform: translateY(30px) scale(0.8);

filter: blur(10px);

}

50% {

opacity: 0.8;

transform: translateY(10px) scale(0.95);

filter: blur(2px);

}

100% {

opacity: 1;

transform: translateY(0) scale(1);

filter: blur(0);

}

}

@keyframes grid-draw {

0% {

stroke-dashoffset: 1000;

opacity: 0;

}

50% {

opacity: 0.3;

}

100% {

stroke-dashoffset: 0;

opacity: 0.15;

}

}

@keyframes pulse-glow {

0%,

100% {

opacity: 0.1;

transform: scale(1);

}

50% {

opacity: 0.3;

transform: scale(1.1);

}

}

@keyframes underline-grow {

to {

width: 100%;

}

}

@keyframes float {

0%,

100% {

transform: translateY(0) translateX(0);

opacity: 0.2;

}

25% {

transform: translateY(-10px) translateX(5px);

opacity: 0.6;

}

50% {

transform: translateY(-5px) translateX(-3px);

opacity: 0.4;

}

75% {

transform: translateY(-15px) translateX(7px);

opacity: 0.8;

}

}