:root {

--color-bg1: rgb(75, 143, 232); /\* Dark blue background \*/

--color-bg2: rgb(48, 151, 248); /\* Lighter blue background \*/

--color1: rgba(102, 51, 153, 0.8); /\* Purple with opacity \*/

--color2: rgba(153, 102, 204, 0.8); /\* Light purple with opacity \*/

--color3: rgba(204, 153, 255, 0.8); /\* Lighter purple with opacity \*/

--color4: rgba(153, 51, 255, 0.8); /\* Vivid purple with opacity \*/

--color5: rgba(100, 149, 237, 0.8); /\* Cornflower blue with opacity \*/

--color-interactive: rgba(211, 211, 211, 0.7); /\* Light Gray with opacity \*/

--circle-size: 80%;

--blending: hard-light;

--title-color: #ffffff; /\* White color for the title \*/

}

.gradient-bg {

position: absolute;

top: 0;

left: 0;

width: 100%;

height: 100vh;

background: linear-gradient(40deg, var(--color-bg1), var(--color-bg2));

z-index: -2; /\* Ensure the background is behind everything \*/

overflow: hidden; /\* Prevent any child elements from overflowing \*/

}

.gradients-container {

width: 100%;

height: 100%;

position: absolute;

top: 0;

left: 0;

z-index: -1; /\* Ensure the circles are above the gradient background but below content \*/

pointer-events: none; /\* Prevent the gradients from interfering with user interactions \*/

}

.g1, .g2, .g3, .g4, .g5 {

position: absolute;

background: radial-gradient(circle at center, var(--color1) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

mix-blend-mode: var(--blending);

width: var(--circle-size);

height: var(--circle-size);

border-radius: 50%;

opacity: 0.8; /\* Ensure visibility \*/

z-index: -1; /\* Ensure these are layered correctly \*/

}

.g1 {

top: calc(50% - var(--circle-size) / 2);

left: calc(50% - var(--circle-size) / 2);

background: radial-gradient(circle at center, var(--color1) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

animation: moveVertical 15s ease-in-out infinite;

}

.g2 {

top: calc(50% - var(--circle-size) / 2);

left: calc(50% - var(--circle-size) / 2);

background: radial-gradient(circle at center, var(--color2) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

animation: moveInCircle 10s ease-in-out infinite;

}

.g3 {

top: calc(50% - var(--circle-size) / 2 + 200px);

left: calc(50% - var(--circle-size) / 2 - 500px);

background: radial-gradient(circle at center, var(--color3) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

animation: moveInCircle 20s linear infinite;

}

.g4 {

top: calc(50% - var(--circle-size) / 2);

left: calc(50% - var(--circle-size) / 2);

background: radial-gradient(circle at center, var(--color4) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

animation: moveHorizontal 25s ease-in infinite;

}

.g5 {

width: calc(var(--circle-size) \* 2);

height: calc(var(--circle-size) \* 2);

top: calc(50% - var(--circle-size));

left: calc(50% - var(--circle-size));

background: radial-gradient(circle at center, var(--color5) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

animation: moveInCircle 15s ease-out infinite;

}

.interactive {

position: absolute;

background: radial-gradient(circle at center, var(--color-interactive) 0%, rgba(255, 255, 255, 0) 50%) no-repeat;

mix-blend-mode: var(--blending);

width: 150px;

height: 150px;

border-radius: 50%;

top: 0;

left: 0;

pointer-events: none; /\* Prevent the circle from interfering with user interactions \*/

z-index: 100; /\* Ensure the circle is on top \*/

}

.text-container {

position: relative;

z-index: 100;

width: 100%;

height: 83vh;

padding: 20px;

color: white;

opacity: 0.9;

user-select: none;

text-shadow: 1px 1px rgba(0, 0, 0, 0.5); /\* Darker shadow for better visibility \*/

display: flex;

flex-direction: column;

justify-content: flex-start; /\* Align content to the top \*/

overflow: hidden;

}

.text-container h1 {

font-family: 'Verdana', sans-serif;

letter-spacing: 0.05em;

text-transform: uppercase;

font-size: 4rem; /\* Adjust font size \*/

font-weight: 700;

text-align: left; /\* Align title to the left \*/

margin: 0 0 20px 0;

padding-top: 10px; /\* Adjust top padding to move the title up \*/

background: linear-gradient(90deg, #333, #666); /\* Subtle, darker tones \*/

-webkit-background-clip: text;

text-shadow: 2px 2px 5px rgba(0, 0, 0, 0.3),

-2px -2px 5px rgba(210, 210, 210, 0.628);

display: inline-block;

animation: slideIn 2s ease-out; /\* Load-in animation \*/

}

.content-wrapper {

display: flex;

flex-direction: row; /\* Align items horizontally \*/

align-items: flex-start; /\* Align items to the top \*/

justify-content: space-between; /\* Space between text and image \*/

max-width: 90%;

margin: 0 auto;

height: 100%;

}

.text-content {

flex: 1;

max-width: 45%; /\* Takes up less than half of the container \*/

margin-top: 60px;

margin-right: 20px; /\* Add space between text and image \*/

display: flex;

flex-direction: column;

justify-content: flex-start; /\* Align content to the top \*/

}

.image-container {

flex: 1;

max-width: 45%; /\* Prevent the container from taking up too much space \*/

background: transparent;

position: relative;

display: flex;

justify-content: center; /\* Center horizontally \*/

align-items: center; /\* Center vertically \*/

overflow: hidden;

height: 500px; /\* Set a fixed height based on the largest image \*/

}

.image-container img {

max-width: 100%; /\* Prevent the image from exceeding the container's width \*/

max-height: 100%; /\* Prevent the image from exceeding the container's height \*/

width: auto; /\* Maintain the image's original aspect ratio \*/

height: auto; /\* Maintain the image's original aspect ratio \*/

transition: opacity 0.5s ease-in-out; /\* Smooth fade transition \*/

position: absolute; /\* Position the image absolutely within the container \*/

top: 50%; /\* Position from the center \*/

left: 50%; /\* Position from the center \*/

transform: translate(-50%, -50%); /\* Center the image both horizontally and vertically \*/

opacity: 0; /\* Start with the image hidden \*/

}

.image-container img.fade-in {

opacity: 1; /\* Fade in the current image \*/

}

.X39-info {

font-family: 'Roboto', sans-serif;

font-size: 1.5rem;

line-height: 1.6;

font-weight: 400;

letter-spacing: 0.5px;

color: #f0f0f0;

background-color: rgba(55, 57, 66, 0.9);

padding: 20px;

border-radius: 10px;

text-shadow: 1px 1px 5px rgba(0, 0, 0, 0.5);

text-align: left; /\* Align text to the left \*/

box-sizing: border-box; /\* Include padding in the width \*/

margin-bottom: 20px; /\* Ensure space below the text for buttons \*/

}

.button-wrapper {

display: flex;

flex-direction: row;

gap: 30px; /\* Adds a 30px gap between each button \*/

justify-content: center; /\* Center the buttons \*/

align-items: center;

left: 5px;

margin-top: 140px; /\* Adjust spacing above the buttons \*/

width: 100%; /\* Ensure the container takes the full width \*/

}

.button-wrapper button {

flex-shrink: 0 /\* Prevent buttons from shrinking \*/

}

@keyframes slideIn {

from {

opacity: 0;

transform: translateY(-20px);

}

to {

opacity: 1;

transform: translateY(0);

}

}

@keyframes animateBorder {

0% {

background-position: 0% 50%;

}

50% {

background-position: 100% 50%;

}

100% {

background-position: 0% 50%;

}

}

@keyframes moveInCircle {

0% {

transform: rotate(0deg);

}

50% {

transform: rotate(100deg);

}

100% {

transform: rotate(360deg);

}

}

@keyframes moveVertical {

0% {

transform: translateY(-50%);

}

50% {

transform: translateY(50%);

}

100% {

transform: translateY(-50%);

}

}

@keyframes moveHorizontal {

0% {

transform: translateX(-50%) translateY(-10%);

}

50% {

transform: translateX(50%) translateY(10%);

}

100% {

transform: translateX(-50%) translateY(-10%);

}

}

.footer {

width: 100%;

padding: 10px;

background-color: #333;

color: white;

text-align: center;

position: relative;

bottom: 0;

left: 0;

font-size: 14px;

z-index: 100;

}

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

import benefitsX39 from "../images/benefitsX39.jpg";

import infoX39 from "../images/infoX39.jpg";

import benefitsblue from "../images/benefitsblue.jpg";

import Button from "./Button";

import "./Homepage.css";

function Homepage() {

const [currentImageIndex, setCurrentImageIndex] = useState(0);

const [fade, setFade] = useState(true);

const images = [benefitsX39, infoX39, benefitsblue];

useEffect(() => {

const intervalId = setInterval(() => {

setFade(false);

setTimeout(() => {

setCurrentImageIndex((prevIndex) => (prevIndex + 1) % images.length);

setFade(true);

}, 500); // Duration of fade out transition

}, 3000); // Change image every 3 seconds

return () => clearInterval(intervalId);

}, [images.length]);

// Mouse movement effect

useEffect(() => {

const interactiveCircle = document.querySelector(".interactive");

let curX = 0;

let curY = 0;

let tgX = 0;

let tgY = 0;

const move = () => {

curX += (tgX - curX) / 10;

curY += (tgY - curY) / 10;

interactiveCircle.style.transform = `translate(${Math.round(curX)}px, ${Math.round(curY)}px)`;

requestAnimationFrame(move);

};

const handleMouseMove = (event) => {

tgX = event.clientX - interactiveCircle.offsetWidth / 2;

tgY = event.clientY - interactiveCircle.offsetHeight / 2;

};

window.addEventListener("mousemove", handleMouseMove);

move();

return () => {

window.removeEventListener("mousemove", handleMouseMove);

};

}, []);

return (

<div>

<div className="text-container">

<h1 className="title">

<i className="fas fa-bolt"></i> <span>The</span> <span>Power</span> <span>of</span> <span>X39</span>

</h1>

<div className="content-wrapper">

<div className="text-content">

<p className="X39-info">

<b>X39</b> is a revolutionary wearable technology designed to

activate your body's natural healing processes, promoting faster

recovery, improved wellness, and a renewed sense of vitality.

</p>

<div className="button-wrapper">

<Button

text="Buy Now"

link="https://lifewave.com/kristinasimms"

variant="primary"

/>

<Button

text="Learn More"

link="https://thisisitinfo.com/"

variant="secondary"

/>

</div>

</div>

<div className="image-container">

<img

src={images[currentImageIndex]}

alt="Benefits of X39"

className={`fade ${fade ? "fade-in" : "fade-out"}`}

/>

</div>

</div>

</div>

<div className="gradient-bg">

<svg xmlns="http://www.w3.org/2000/svg">

<defs>

<filter id="goo">

<feGaussianBlur

in="SourceGraphic"

stdDeviation="10"

result="blur"

/>

<feColorMatrix

in="blur"

mode="matrix"

values="1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 18 -8"

result="goo"

/>

<feBlend in="SourceGraphic" in2="goo" />

</filter>

</defs>

</svg>

<div className="gradients-container">

<div className="g1"></div>

<div className="g2"></div>

<div className="g3"></div>

<div className="g4"></div>

<div className="g5"></div>

<div className="interactive"></div>

</div>

</div>

</div>

);

}

export default Homepage;