Build this as my initial prototype

Copy-paste this component to /components/ui folder:

```tsx

sign-in-card-2.tsx

'use client'

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

import Link from 'next/link';

import { motion, AnimatePresence, useMotionValue, useTransform } from 'framer-motion';

import { Mail, Lock,Eye,EyeClosed, ArrowRight } from 'lucide-react';

import { cn } from "@/lib/utils"

function Input({ className, type, ...props }: React.ComponentProps<"input">) {

return (

<input

type={type}

data-slot="input"

className={cn(

"file:text-foreground placeholder:text-muted-foreground selection:bg-primary selection:text-primary-foreground dark:bg-input/30 border-input flex h-9 w-full min-w-0 rounded-md border bg-transparent px-3 py-1 text-base shadow-xs transition-[color,box-shadow] outline-none file:inline-flex file:h-7 file:border-0 file:bg-transparent file:text-sm file:font-medium disabled:pointer-events-none disabled:cursor-not-allowed disabled:opacity-50 md:text-sm",

"focus-visible:border-ring focus-visible:ring-ring/50 focus-visible:ring-[3px]",

"aria-invalid:ring-destructive/20 dark:aria-invalid:ring-destructive/40 aria-invalid:border-destructive",

className

)}

{...props}

/>

)

}

export function Component() {

const [showPassword, setShowPassword] = useState(false);

const [email, setEmail] = useState("");

const [password, setPassword] = useState("");

const [isLoading, setIsLoading] = useState(false);

const [focusedInput, setFocusedInput] = useState(null);

const [rememberMe, setRememberMe] = useState(false);

const [mousePosition, setMousePosition] = useState({ x: 0, y: 0 });

// For 3D card effect - increased rotation range for more pronounced 3D effect

const mouseX = useMotionValue(0);

const mouseY = useMotionValue(0);

const rotateX = useTransform(mouseY, [-300, 300], [10, -10]); // Increased from 5/-5 to 10/-10

const rotateY = useTransform(mouseX, [-300, 300], [-10, 10]); // Increased from -5/5 to -10/10

const handleMouseMove = (e: React.MouseEvent) => {

const rect = e.currentTarget.getBoundingClientRect();

mouseX.set(e.clientX - rect.left - rect.width / 2);

mouseY.set(e.clientY - rect.top - rect.height / 2);

setMousePosition({ x: e.clientX, y: e.clientY });

};

const handleMouseLeave = () => {

mouseX.set(0);

mouseY.set(0);

};

const handleSubmit = (event: React.MouseEvent) => {

event.preventDefault();

setIsLoading(true);

setTimeout(() => setIsLoading(false), 2000);

};

return (

<div className="min-h-screen w-screen bg-black relative overflow-hidden flex items-center justify-center">

{/\* Background gradient effect - matches the purple OnlyPipe style \*/}

<div className="absolute inset-0 bg-gradient-to-b from-purple-500/40 via-purple-700/50 to-black" />

{/\* Subtle noise texture overlay \*/}

<div className="absolute inset-0 opacity-[0.03] mix-blend-soft-light"

style={{

backgroundImage: `url("data:image/svg+xml,%3Csvg viewBox='0 0 200 200' xmlns='http://www.w3.org/2000/svg'%3E%3Cfilter id='noiseFilter'%3E%3CfeTurbulence type='fractalNoise' baseFrequency='0.65' numOctaves='3' stitchTiles='stitch'/%3E%3C/filter%3E%3Crect width='100%25' height='100%25' filter='url(%23noiseFilter)'/%3E%3C/svg%3E")`,

backgroundSize: '200px 200px'

}}

/>

{/\* Top radial glow \*/}

<div className="absolute top-0 left-1/2 transform -translate-x-1/2 w-[120vh] h-[60vh] rounded-b-[50%] bg-purple-400/20 blur-[80px]" />

<motion.div

className="absolute top-0 left-1/2 transform -translate-x-1/2 w-[100vh] h-[60vh] rounded-b-full bg-purple-300/20 blur-[60px]"

animate={{

opacity: [0.15, 0.3, 0.15],

scale: [0.98, 1.02, 0.98]

}}

transition={{

duration: 8,

repeat: Infinity,

repeatType: "mirror"

}}

/>

<motion.div

className="absolute bottom-0 left-1/2 transform -translate-x-1/2 w-[90vh] h-[90vh] rounded-t-full bg-purple-400/20 blur-[60px]"

animate={{

opacity: [0.3, 0.5, 0.3],

scale: [1, 1.1, 1]

}}

transition={{

duration: 6,

repeat: Infinity,

repeatType: "mirror",

delay: 1

}}

/>

{/\* Animated glow spots \*/}

<div className="absolute left-1/4 top-1/4 w-96 h-96 bg-white/5 rounded-full blur-[100px] animate-pulse opacity-40" />

<div className="absolute right-1/4 bottom-1/4 w-96 h-96 bg-white/5 rounded-full blur-[100px] animate-pulse delay-1000 opacity-40" />

<motion.div

initial={{ opacity: 0, y: 20 }}

animate={{ opacity: 1, y: 0 }}

transition={{ duration: 0.8 }}

className="w-full max-w-sm relative z-10"

style={{ perspective: 1500 }}

>

<motion.div

className="relative"

style={{ rotateX, rotateY }}

onMouseMove={handleMouseMove}

onMouseLeave={handleMouseLeave}

whileHover={{ z: 10 }}

>

<div className="relative group">

{/\* Card glow effect - reduced intensity \*/}

<motion.div

className="absolute -inset-[1px] rounded-2xl opacity-0 group-hover:opacity-70 transition-opacity duration-700"

animate={{

boxShadow: [

"0 0 10px 2px rgba(255,255,255,0.03)",

"0 0 15px 5px rgba(255,255,255,0.05)",

"0 0 10px 2px rgba(255,255,255,0.03)"

],

opacity: [0.2, 0.4, 0.2]

}}

transition={{

duration: 4,

repeat: Infinity,

ease: "easeInOut",

repeatType: "mirror"

}}

/>

{/\* Traveling light beam effect - reduced opacity \*/}

<div className="absolute -inset-[1px] rounded-2xl overflow-hidden">

{/\* Top light beam - enhanced glow \*/}

<motion.div

className="absolute top-0 left-0 h-[3px] w-[50%] bg-gradient-to-r from-transparent via-white to-transparent opacity-70"

initial={{ filter: "blur(2px)" }}

animate={{

left: ["-50%", "100%"],

opacity: [0.3, 0.7, 0.3],

filter: ["blur(1px)", "blur(2.5px)", "blur(1px)"]

}}

transition={{

left: {

duration: 2.5,

ease: "easeInOut",

repeat: Infinity,

repeatDelay: 1

},

opacity: {

duration: 1.2,

repeat: Infinity,

repeatType: "mirror"

},

filter: {

duration: 1.5,

repeat: Infinity,

repeatType: "mirror"

}

}}

/>

{/\* Right light beam - enhanced glow \*/}

<motion.div

className="absolute top-0 right-0 h-[50%] w-[3px] bg-gradient-to-b from-transparent via-white to-transparent opacity-70"

initial={{ filter: "blur(2px)" }}

animate={{

top: ["-50%", "100%"],

opacity: [0.3, 0.7, 0.3],

filter: ["blur(1px)", "blur(2.5px)", "blur(1px)"]

}}

transition={{

top: {

duration: 2.5,

ease: "easeInOut",

repeat: Infinity,

repeatDelay: 1,

delay: 0.6

},

opacity: {

duration: 1.2,

repeat: Infinity,

repeatType: "mirror",

delay: 0.6

},

filter: {

duration: 1.5,

repeat: Infinity,

repeatType: "mirror",

delay: 0.6

}

}}

/>

{/\* Bottom light beam - enhanced glow \*/}

<motion.div

className="absolute bottom-0 right-0 h-[3px] w-[50%] bg-gradient-to-r from-transparent via-white to-transparent opacity-70"

initial={{ filter: "blur(2px)" }}

animate={{

right: ["-50%", "100%"],

opacity: [0.3, 0.7, 0.3],

filter: ["blur(1px)", "blur(2.5px)", "blur(1px)"]

}}

transition={{

right: {

duration: 2.5,

ease: "easeInOut",

repeat: Infinity,

repeatDelay: 1,

delay: 1.2

},

opacity: {

duration: 1.2,

repeat: Infinity,

repeatType: "mirror",

delay: 1.2

},

filter: {

duration: 1.5,

repeat: Infinity,

repeatType: "mirror",

delay: 1.2

}

}}

/>

{/\* Left light beam - enhanced glow \*/}

<motion.div

className="absolute bottom-0 left-0 h-[50%] w-[3px] bg-gradient-to-b from-transparent via-white to-transparent opacity-70"

initial={{ filter: "blur(2px)" }}

animate={{

bottom: ["-50%", "100%"],

opacity: [0.3, 0.7, 0.3],

filter: ["blur(1px)", "blur(2.5px)", "blur(1px)"]

}}

transition={{

bottom: {

duration: 2.5,

ease: "easeInOut",

repeat: Infinity,

repeatDelay: 1,

delay: 1.8

},

opacity: {

duration: 1.2,

repeat: Infinity,

repeatType: "mirror",

delay: 1.8

},

filter: {

duration: 1.5,

repeat: Infinity,

repeatType: "mirror",

delay: 1.8

}

}}

/>

{/\* Subtle corner glow spots - reduced opacity \*/}

<motion.div

className="absolute top-0 left-0 h-[5px] w-[5px] rounded-full bg-white/40 blur-[1px]"

animate={{

opacity: [0.2, 0.4, 0.2]

}}

transition={{

duration: 2,

repeat: Infinity,

repeatType: "mirror"

}}

/>

<motion.div

className="absolute top-0 right-0 h-[8px] w-[8px] rounded-full bg-white/60 blur-[2px]"

animate={{

opacity: [0.2, 0.4, 0.2]

}}

transition={{

duration: 2.4,

repeat: Infinity,

repeatType: "mirror",

delay: 0.5

}}

/>

<motion.div

className="absolute bottom-0 right-0 h-[8px] w-[8px] rounded-full bg-white/60 blur-[2px]"

animate={{

opacity: [0.2, 0.4, 0.2]

}}

transition={{

duration: 2.2,

repeat: Infinity,

repeatType: "mirror",

delay: 1

}}

/>

<motion.div

className="absolute bottom-0 left-0 h-[5px] w-[5px] rounded-full bg-white/40 blur-[1px]"

animate={{

opacity: [0.2, 0.4, 0.2]

}}

transition={{

duration: 2.3,

repeat: Infinity,

repeatType: "mirror",

delay: 1.5

}}

/>

</div>

{/\* Card border glow - reduced opacity \*/}

<div className="absolute -inset-[0.5px] rounded-2xl bg-gradient-to-r from-white/3 via-white/7 to-white/3 opacity-0 group-hover:opacity-70 transition-opacity duration-500" />

{/\* Glass card background \*/}

<div className="relative bg-black/40 backdrop-blur-xl rounded-2xl p-6 border border-white/[0.05] shadow-2xl overflow-hidden">

{/\* Subtle card inner patterns \*/}

<div className="absolute inset-0 opacity-[0.03]"

style={{

backgroundImage: `linear-gradient(135deg, white 0.5px, transparent 0.5px), linear-gradient(45deg, white 0.5px, transparent 0.5px)`,

backgroundSize: '30px 30px'

}}

/>

{/\* Logo and header \*/}

<div className="text-center space-y-1 mb-5">

<motion.div

initial={{ scale: 0.5, opacity: 0 }}

animate={{ scale: 1, opacity: 1 }}

transition={{ type: "spring", duration: 0.8 }}

className="mx-auto w-10 h-10 rounded-full border border-white/10 flex items-center justify-center relative overflow-hidden"

>

{/\* Logo placeholder - would be an SVG in practice \*/}

{/\* <!-- SVG\_LOGO --> \*/}

<span className="text-lg font-bold bg-clip-text text-transparent bg-gradient-to-b from-white to-white/70">S</span>

{/\* Inner lighting effect \*/}

<div className="absolute inset-0 bg-gradient-to-br from-white/10 to-transparent opacity-50" />

</motion.div>

<motion.h1

initial={{ opacity: 0, y: 10 }}

animate={{ opacity: 1, y: 0 }}

transition={{ delay: 0.2 }}

className="text-xl font-bold bg-clip-text text-transparent bg-gradient-to-b from-white to-white/80"

>

Welcome Back

</motion.h1>

<motion.p

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

transition={{ delay: 0.3 }}

className="text-white/60 text-xs"

>

Sign in to continue to StyleMe

</motion.p>

</div>

{/\* Login form \*/}

<form onSubmit={(e) => { e.preventDefault(); setIsLoading(true); setTimeout(() => setIsLoading(false), 2000); }} className="space-y-4">

<motion.div className="space-y-3">

{/\* Email input \*/}

<motion.div

className={`relative ${focusedInput === "email" ? 'z-10' : ''}`}

whileFocus={{ scale: 1.02 }}

whileHover={{ scale: 1.01 }}

transition={{ type: "spring", stiffness: 400, damping: 25 }}

>

<div className="absolute -inset-[0.5px] bg-gradient-to-r from-white/10 via-white/5 to-white/10 rounded-lg opacity-0 group-hover:opacity-100 transition-all duration-300" />

<div className="relative flex items-center overflow-hidden rounded-lg">

<Mail className={`absolute left-3 w-4 h-4 transition-all duration-300 ${

focusedInput === "email" ? 'text-white' : 'text-white/40'

}`} />

<Input

type="email"

placeholder="Email address"

value={email}

onChange={(e) => setEmail(e.target.value)}

onFocus={() => setFocusedInput("email" as any)}

onBlur={() => setFocusedInput(null)}

className="w-full bg-white/5 border-transparent focus:border-white/20 text-white placeholder:text-white/30 h-10 transition-all duration-300 pl-10 pr-3 focus:bg-white/10"

/>

{/\* Input highlight effect \*/}

{focusedInput === "email" && (

<motion.div

layoutId="input-highlight"

className="absolute inset-0 bg-white/5 -z-10"

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

exit={{ opacity: 0 }}

transition={{ duration: 0.2 }}

/>

)}

</div>

</motion.div>

{/\* Password input \*/}

<motion.div

className={`relative ${focusedInput === "password" ? 'z-10' : ''}`}

whileFocus={{ scale: 1.02 }}

whileHover={{ scale: 1.01 }}

transition={{ type: "spring", stiffness: 400, damping: 25 }}

>

<div className="absolute -inset-[0.5px] bg-gradient-to-r from-white/10 via-white/5 to-white/10 rounded-lg opacity-0 group-hover:opacity-100 transition-all duration-300" />

<div className="relative flex items-center overflow-hidden rounded-lg">

<Lock className={`absolute left-3 w-4 h-4 transition-all duration-300 ${

focusedInput === "password" ? 'text-white' : 'text-white/40'

}`} />

<Input

type={showPassword ? "text" : "password"}

placeholder="Password"

value={password}

onChange={(e) => setPassword(e.target.value)}

onFocus={() => setFocusedInput("password" as any)}

onBlur={() => setFocusedInput(null)}

className="w-full bg-white/5 border-transparent focus:border-white/20 text-white placeholder:text-white/30 h-10 transition-all duration-300 pl-10 pr-10 focus:bg-white/10"

/>

{/\* Toggle password visibility \*/}

<div

onClick={() => setShowPassword(!showPassword)}

className="absolute right-3 cursor-pointer"

>

{showPassword ? (

<Eye className="w-4 h-4 text-white/40 hover:text-white transition-colors duration-300" />

) : (

<EyeClosed className="w-4 h-4 text-white/40 hover:text-white transition-colors duration-300" />

)}

</div>

{/\* Input highlight effect \*/}

{focusedInput === "password" && (

<motion.div

layoutId="input-highlight"

className="absolute inset-0 bg-white/5 -z-10"

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

exit={{ opacity: 0 }}

transition={{ duration: 0.2 }}

/>

)}

</div>

</motion.div>

</motion.div>

{/\* Remember me & Forgot password \*/}

<div className="flex items-center justify-between pt-1">

<div className="flex items-center space-x-2">

<div className="relative">

<input

id="remember-me"

name="remember-me"

type="checkbox"

checked={rememberMe}

onChange={() => setRememberMe(!rememberMe)}

className="appearance-none h-4 w-4 rounded border border-white/20 bg-white/5 checked:bg-white checked:border-white focus:outline-none focus:ring-1 focus:ring-white/30 transition-all duration-200"

/>

{rememberMe && (

<motion.div

initial={{ opacity: 0, scale: 0.5 }}

animate={{ opacity: 1, scale: 1 }}

className="absolute inset-0 flex items-center justify-center text-black pointer-events-none"

>

{/\* <!-- SVG\_CHECKMARK --> \*/}

<svg xmlns="http://www.w3.org/2000/svg" width="12" height="12" viewBox="0 0 24 24" fill="none" stroke="currentColor" strokeWidth="3" strokeLinecap="round" strokeLinejoin="round">

<polyline points="20 6 9 17 4 12"></polyline>

</svg>

</motion.div>

)}

</div>

<label htmlFor="remember-me" className="text-xs text-white/60 hover:text-white/80 transition-colors duration-200">

Remember me

</label>

</div>

<div className="text-xs relative group/link">

<Link href="/forgot-password" className="text-white/60 hover:text-white transition-colors duration-200">

Forgot password?

</Link>

</div>

</div>

{/\* Sign in button \*/}

<motion.button

whileHover={{ scale: 1.02 }}

whileTap={{ scale: 0.98 }}

type="submit"

disabled={isLoading}

className="w-full relative group/button mt-5"

>

{/\* Button glow effect - reduced intensity \*/}

<div className="absolute inset-0 bg-white/10 rounded-lg blur-lg opacity-0 group-hover/button:opacity-70 transition-opacity duration-300" />

<div className="relative overflow-hidden bg-white text-black font-medium h-10 rounded-lg transition-all duration-300 flex items-center justify-center">

{/\* Button background animation \*/}

<motion.div

className="absolute inset-0 bg-gradient-to-r from-white/0 via-white/30 to-white/0 -z-10"

animate={{

x: ['-100%', '100%'],

}}

transition={{

duration: 1.5,

ease: "easeInOut",

repeat: Infinity,

repeatDelay: 1

}}

style={{

opacity: isLoading ? 1 : 0,

transition: 'opacity 0.3s ease'

}}

/>

<AnimatePresence mode="wait">

{isLoading ? (

<motion.div

key="loading"

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

exit={{ opacity: 0 }}

className="flex items-center justify-center"

>

<div className="w-4 h-4 border-2 border-black/70 border-t-transparent rounded-full animate-spin" />

</motion.div>

) : (

<motion.span

key="button-text"

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

exit={{ opacity: 0 }}

className="flex items-center justify-center gap-1 text-sm font-medium"

>

Sign In

<ArrowRight className="w-3 h-3 group-hover/button:translate-x-1 transition-transform duration-300" />

</motion.span>

)}

</AnimatePresence>

</div>

</motion.button>

{/\* Minimal Divider \*/}

<div className="relative mt-2 mb-5 flex items-center">

<div className="flex-grow border-t border-white/5"></div>

<motion.span

className="mx-3 text-xs text-white/40"

initial={{ opacity: 0.7 }}

animate={{ opacity: [0.7, 0.9, 0.7] }}

transition={{ duration: 3, repeat: Infinity, ease: "easeInOut" }}

>

or

</motion.span>

<div className="flex-grow border-t border-white/5"></div>

</div>

{/\* Google Sign In \*/}

<motion.button

whileHover={{ scale: 1.02 }}

whileTap={{ scale: 0.98 }}

type="button"

className="w-full relative group/google"

>

<div className="absolute inset-0 bg-white/5 rounded-lg blur opacity-0 group-hover/google:opacity-70 transition-opacity duration-300" />

<div className="relative overflow-hidden bg-white/5 text-white font-medium h-10 rounded-lg border border-white/10 hover:border-white/20 transition-all duration-300 flex items-center justify-center gap-2">

{/\* <!-- SVG\_GOOGLE\_LOGO --> \*/}

<div className="w-4 h-4 flex items-center justify-center text-white/80 group-hover/google:text-white transition-colors duration-300">G</div>

<span className="text-white/80 group-hover/google:text-white transition-colors text-xs">

Sign in with Google

</span>

{/\* Button hover effect \*/}

<motion.div

className="absolute inset-0 bg-gradient-to-r from-white/0 via-white/5 to-white/0"

initial={{ x: '-100%' }}

whileHover={{ x: '100%' }}

transition={{

duration: 1,

ease: "easeInOut"

}}

/>

</div>

</motion.button>

{/\* Sign up link \*/}

<motion.p

className="text-center text-xs text-white/60 mt-4"

initial={{ opacity: 0 }}

animate={{ opacity: 1 }}

transition={{ delay: 0.5 }}

>

Don't have an account?{' '}

<Link

href="/signup"

className="relative inline-block group/signup"

>

<span className="relative z-10 text-white group-hover/signup:text-white/70 transition-colors duration-300 font-medium">

Sign up

</span>

<span className="absolute bottom-0 left-0 w-0 h-[1px] bg-white group-hover/signup:w-full transition-all duration-300" />

</Link>

</motion.p>

</form>

</div>

</div>

</motion.div>

</motion.div>

</div>

);

}

demo.tsx

import { Component } from "@/components/ui/sign-in-card-2";

const DemoOne = () => {

return (

<div className="flex w-full h-screen justify-center items-center">

<Component />

</div>

);

};

export { DemoOne };

```

Install these NPM dependencies:

```bash

next, lucide-react, framer-motion

```