

Build this as my initial prototype

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

```
``tsx
sign-in.tsx
import React, { useState } from 'react';
import { Eye, EyeOff } from 'lucide-react';

// --- HELPER COMPONENTS (ICONS) ---

const GoogleIcon = () => (
  <svg xmlns="http://www.w3.org/2000/svg" className="h-5 w-5" viewBox="0 0 48 48">
    <path fill="#FFC107" d="M43.611 20.083H20H24v8h11.303c-1.649 4.657-6.08
8-11.303 8-6.627 0-12-5.373-12-12s12-5.373 12-12c3.059 0 5.842 1.154 7.961
3.039l5.657-5.657C34.046 6.053 29.268 4 24 4 12.955 4 12.955 4 24s8.955 20 20
20-8.955 20-20c0-2.641-.21-5.236-.611-7.743z" />
    <path fill="#FF3D00" d="M6.306 14.691l6.571 4.819C14.655 15.108 18.961 12 24
12c3.059 0 5.842 1.154 7.961 3.039l5.657-5.657C34.046 6.053 29.268 4 24 4 16.318 4 9.656
8.337 6.306 14.691z" />
    <path fill="#4CAF50" d="M24 44c5.166 0 9.86-1.977 13.409-5.192l-6.19-5.238C29.211
35.091 26.715 36 24 36c-5.202 0-9.619-3.317-11.283-7.946l-6.522 5.025C9.505 39.556 16.227
44 24 44z" />
    <path fill="#1976D2" d="M43.611 20.083H20H24v8h11.303c-.792 2.237-2.231
4.166-4.087 5.571l6.19 5.238C42.022 35.026 44 30.038 44 24c0-2.641-.21-5.236-.611-7.743z"
  />
  </svg>
);

// --- TYPE DEFINITIONS ---

export interface Testimonial {
  avatarSrc: string;
  name: string;
  handle: string;
  text: string;
}

interface SignInPageProps {
  title?: React.ReactNode;
  description?: React.ReactNode;
  heroImageSrc?: string;
  testimonials?: Testimonial[];
  onSignIn?: (event: React.FormEvent<HTMLFormElement>) => void;
```

```

onGoogleSignIn?: () => void;
onResetPassword?: () => void;
onCreateAccount?: () => void;
}

// --- SUB-COMPONENTS ---

const GlassInputWrapper = ({ children }: { children: React.ReactNode }) => (
  <div className="rounded-2xl border border-border bg-foreground/5 backdrop-blur-sm
transition-colors focus-within:border-violet-400/70 focus-within:bg-violet-500/10">
    {children}
  </div>
);

const TestimonialCard = ({ testimonial, delay }: { testimonial: Testimonial, delay: string }) => (
  <div className={`animate-testimonial ${delay} flex items-start gap-3 rounded-3xl bg-card/40
dark:bg-zinc-800/40 backdrop-blur-xl border border-white/10 p-5 w-64`}>
    <img src={testimonial.avatarSrc} className="h-10 w-10 object-cover rounded-2xl"
alt="avatar" />
    <div className="text-sm leading-snug">
      <p className="flex items-center gap-1 font-medium">{testimonial.name}</p>
      <p className="text-muted-foreground">{testimonial.handle}</p>
      <p className="mt-1 text-foreground/80">{testimonial.text}</p>
    </div>
  </div>
);

// --- MAIN COMPONENT ---

export const SignInPage: React.FC<SignInPageProps> = ({
  title = <span className="font-light text-foreground tracking-tighter">Welcome</span>,
  description = "Access your account and continue your journey with us",
  heroImageSrc,
  testimonials = [],
  onSignIn,
  onGoogleSignIn,
  onResetPassword,
  onCreateAccount,
}) => {
  const [showPassword, setShowPassword] = useState(false);

  return (
    <div className="h-[100dvh] flex flex-col md:flex-row font-geist w-[100dvw]">
      {/* Left column: sign-in form */}

```

```

<section className="flex-1 flex items-center justify-center p-8">
  <div className="w-full max-w-md">
    <div className="flex flex-col gap-6">
      <h1 className="animate-element animate-delay-100 text-4xl md:text-5xl font-semibold
leading-tight">{title}</h1>
      <p className="animate-element animate-delay-200
text-muted-foreground">{description}</p>

      <form className="space-y-5" onSubmit={onSignIn}>
        <div className="animate-element animate-delay-300">
          <label className="text-sm font-medium text-muted-foreground">Email
Address</label>
          <GlassInputWrapper>
            <input name="email" type="email" placeholder="Enter your email address"
className="w-full bg-transparent text-sm p-4 rounded-2xl focus:outline-none" />
          </GlassInputWrapper>
        </div>

        <div className="animate-element animate-delay-400">
          <label className="text-sm font-medium text-muted-foreground">Password</label>
          <GlassInputWrapper>
            <div className="relative">
              <input name="password" type={showPassword ? 'text' : 'password'}
placeholder="Enter your password" className="w-full bg-transparent text-sm p-4 pr-12
rounded-2xl focus:outline-none" />
              <button type="button" onClick={() => setShowPassword(!showPassword)}
className="absolute inset-y-0 right-3 flex items-center">
                {showPassword ? <EyeOff className="w-5 h-5 text-muted-foreground
hover:text-foreground transition-colors" /> : <Eye className="w-5 h-5 text-muted-foreground
hover:text-foreground transition-colors" />}
              </button>
            </div>
          </GlassInputWrapper>
        </div>

        <div className="animate-element animate-delay-500 flex items-center justify-between
text-sm">
          <label className="flex items-center gap-3 cursor-pointer">
            <input type="checkbox" name="rememberMe" className="custom-checkbox" />
            <span className="text-foreground/90">Keep me signed in</span>
          </label>
          <a href="#" onClick={(e) => { e.preventDefault(); onResetPassword?.(); }}
className="hover:underline text-violet-400 transition-colors">Reset password</a>
        </div>
    </div>
  </div>

```

```

        <button type="submit" className="animate-element animate-delay-600 w-full
rounded-2xl bg-primary py-4 font-medium text-primary-foreground hover:bg-primary/90
transition-colors">
            Sign In
        </button>
    </form>

```

```

    <div className="animate-element animate-delay-700 relative flex items-center
justify-center">
        <span className="w-full border-t border-border"></span>
        <span className="px-4 text-sm text-muted-foreground bg-background absolute">Or
continue with</span>
    </div>

```

```

        <button onClick={onGoogleSignIn} className="animate-element animate-delay-800
w-full flex items-center justify-center gap-3 border border-border rounded-2xl py-4
hover:bg-secondary transition-colors">
            <GoogleIcon />
            Continue with Google
        </button>

```

```

    <p className="animate-element animate-delay-900 text-center text-sm
text-muted-foreground">
        New to our platform? <a href="#" onClick={(e) => { e.preventDefault();
onCreateAccount?.(); }} className="text-violet-400 hover:underline transition-colors">Create
Account</a>
    </p>
</div>
</div>
</section>

```

```

    {/* Right column: hero image + testimonials */}
    {heroImageSrc && (
        <section className="hidden md:block flex-1 relative p-4">
            <div className="animate-slide-right animate-delay-300 absolute inset-4 rounded-3xl
bg-cover bg-center" style={{ backgroundImage: `url(${heroImageSrc})` }}></div>
            {testimonials.length > 0 && (
                <div className="absolute bottom-8 left-1/2 -translate-x-1/2 flex gap-4 px-8 w-full
justify-center">
                    <TestimonialCard testimonial={testimonials[0]} delay="animate-delay-1000" />
                    {testimonials[1] && <div className="hidden xl:flex"><TestimonialCard
testimonial={testimonials[1]} delay="animate-delay-1200" /></div>}
                </div>
            )}
        </section>
    )}

```

```

      {testimonials[2] && <div className="hidden 2xl:flex"><TestimonialCard
testimonial={testimonials[2]} delay="animate-delay-1400" /></div>}
    </div>
  )}
</section>
)}
</div>
);
};

```

demo.tsx

```
import { SignInPage, Testimonial } from "@components/ui/sign-in";
```

```

const sampleTestimonials: Testimonial[] = [
  {
    avatarSrc: "https://randomuser.me/api/portraits/women/57.jpg",
    name: "Sarah Chen",
    handle: "@sarahdigital",
    text: "Amazing platform! The user experience is seamless and the features are exactly what I
needed."
  },
  {
    avatarSrc: "https://randomuser.me/api/portraits/men/64.jpg",
    name: "Marcus Johnson",
    handle: "@marcustech",
    text: "This service has transformed how I work. Clean design, powerful features, and
excellent support."
  },
  {
    avatarSrc: "https://randomuser.me/api/portraits/men/32.jpg",
    name: "David Martinez",
    handle: "@davidcreates",
    text: "I've tried many platforms, but this one stands out. Intuitive, reliable, and genuinely
helpful for productivity."
  },
];

```

```

const SignInPageDemo = () => {
  const handleSignIn = (event: React.FormEvent<HTMLFormElement>) => {
    event.preventDefault();
    const formData = new FormData(event.currentTarget);
    const data = Object.fromEntries(formData.entries());
    console.log("Sign In submitted:", data);
    alert(`Sign In Submitted! Check the browser console for form data.`);
  };
};

```

```

};

const handleGoogleSignIn = () => {
  console.log("Continue with Google clicked");
  alert("Continue with Google clicked");
};

const handleResetPassword = () => {
  alert("Reset Password clicked");
}

const handleCreateAccount = () => {
  alert("Create Account clicked");
}

return (
  <div className="bg-background text-foreground">
    <SignInPage
      heroImageSrc="https://images.unsplash.com/photo-1642615835477-d303d7dc9ee9?w=2160&q=80"
      testimonials={sampleTestimonials}
      onSignIn={handleSignIn}
      onGoogleSignIn={handleGoogleSignIn}
      onResetPassword={handleResetPassword}
      onCreateAccount={handleCreateAccount}
    />
  </div>
);
};

export default SignInPageDemo;

```

Install these NPM dependencies:

```

```bash
lucide-react
```

```

Extend existing Tailwind 4 index.css with this code (or if project uses Tailwind 3, extend tailwind.config.js or globals.css):

```

```css
@import "tailwindcss";
@import "tw-animate-css";

```

```
@keyframes fadeSlideIn {
 to {
 opacity: 1;
 filter: blur(0px);
 transform: translateY(0px);
 }
}
```

```
@keyframes slideRightIn {
 to {
 opacity: 1;
 filter: blur(0px);
 transform: translateX(0px);
 }
}
```

```
@keyframes testimonialIn {
 to {
 opacity: 1;
 filter: blur(0px);
 transform: translateY(0px) scale(1);
 }
}
...
```