

File: client/src/components/CeremonyAnimation.tsx

Type: tsx | Size: 5224 bytes

```
import React, { useEffect, useState } from 'react';
import { motion, AnimatePresence } from 'framer-motion';

interface CeremonyAnimationProps {
  isVisible: boolean;
}

export default function CeremonyAnimation({ isVisible }: CeremonyAnimationProps) {
  const [stars, setStars] = useState<{ id: number; x: number; y: number; delay: number; size: number }[]>([]);

  useEffect(() => {
    if (isVisible) {
      // Generate stars
      const newStars = [];
      for (let i = 0; i < 50; i++) {
        newStars.push({
          id: i,
          x: Math.random() * 100,
          y: Math.random() * 100,
          delay: Math.random() * 2,
          size: Math.random() * 3 + 1
        });
      }
      setStars(newStars);
    }
  }, [isVisible]);

  if (!isVisible) return null;

  return (
    <AnimatePresence>
      {isVisible && (
        <motion.div
          className="fixed inset-0 z-50 flex items-center justify-center bg-mystical-deep/95"
          initial={{ opacity: 0 }}
          animate={{ opacity: 1 }}
          exit={{ opacity: 0 }}
          transition={{ duration: 0.5 }}
        >
          {/* Stars */}
          <div className="absolute inset-0 overflow-hidden">
            {stars.map(star => (
              <motion.div
                key={star.id}
                className="absolute rounded-full bg-white"
                style={{
                  left: `${star.x}%`,
                  top: `${star.y}%`,
                  width: `${star.size}px`,
                  height: `${star.size}px`
                }}
                initial={{ opacity: 0, scale: 0 }}
                animate={{
                  opacity: [0, 1, 0],
                  scale: [0, 1, 0]
                }}
                transition={{
                  duration: 2,
                  delay: star.delay,
                  repeat: Infinity,
                  repeatType: "loop"
                }}
              />
            ))}
          </div>

          {/* Center content */}
          <div className="relative z-10 text-center">
            <motion.div
              className="mb-6 relative"
            />
          </div>
        </div>
      )}
    </AnimatePresence>
  );
}
```

```

    initial={{ scale: 0, rotate: -180 }}
    animate={{ scale: 1, rotate: 0 }}
    transition={{ duration: 1.5, type: "spring" }}
  >
  <div className="w-32 h-32 rounded-full bg-mystical-gradient opacity-20 absolute top-1/2 left-1/2...
  <div className="w-24 h-24 rounded-full bg-mystical-gradient opacity-40 absolute top-1/2 left-1/2...
  <div className="w-20 h-20 rounded-full bg-mystical-gradient flex items-center justify-center rel...
    <i className="fas fa-moon text-4xl text-white"></i>
  </div>
</motion.div>

<motion.h2
  className="text-3xl font-serif text-white mb-2"
  initial={{ opacity: 0, y: 20 }}
  animate={{ opacity: 1, y: 0 }}
  transition={{ delay: 0.5, duration: 0.8 }}
  >
  Aligning the Stars
</motion.h2>

<motion.p
  className="text-mystical-lavender"
  initial={{ opacity: 0 }}
  animate={{ opacity: 1 }}
  transition={{ delay: 1, duration: 0.8 }}
  >
  Drawing your oracle cards...
</motion.p>

<motion.div
  className="mt-8"
  initial={{ opacity: 0, scale: 0.8 }}
  animate={{ opacity: 1, scale: 1 }}
  transition={{ delay: 1.5, duration: 0.8 }}
  >
  <div className="flex justify-center space-x-4">
    <motion.div
      className="w-4 h-4 bg-mystical-pink rounded-full"
      animate={{
        scale: [1, 1.5, 1],
        opacity: [1, 0.5, 1]
      }}
      transition={{
        duration: 1.2,
        repeat: Infinity,
        repeatType: "loop",
        delay: 0
      }}
    />
    <motion.div
      className="w-4 h-4 bg-mystical-purple rounded-full"
      animate={{
        scale: [1, 1.5, 1],
        opacity: [1, 0.5, 1]
      }}
      transition={{
        duration: 1.2,
        repeat: Infinity,
        repeatType: "loop",
        delay: 0.4
      }}
    />
    <motion.div
      className="w-4 h-4 bg-mystical-lavender rounded-full"
      animate={{
        scale: [1, 1.5, 1],
        opacity: [1, 0.5, 1]
      }}
      transition={{
        duration: 1.2,
        repeat: Infinity,
        repeatType: "loop",
        delay: 0.8
      }}
    />
  </div>
</motion.div>

```

```
        }}
      />
    </div>
  </motion.div>
</div>
</motion.div>
)}
</AnimatePresence>
);
}
```