

```
// src/components/Cards/CardDetail.jsx
import { useState, useEffect } from 'react';
import { useSearchParams, Link } from 'react-router-dom';
import { ArrowLeft } from 'lucide-react';
import { useCards } from '../../hooks/useCards';
import AudioPlayer from '../Music/AudioPlayer';

export default function CardDetail() {
  const [searchParams] = useSearchParams();
  const cardId = searchParams.get('id');
  const { getCardById } = useCards();
  const [card, setCard] = useState(null);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    if (!cardId) {
      setLoading(false);
      return;
    }

    const fetchCard = async () => {
      try {
        const data = await getCardById(cardId);
        setCard(data);
      } catch (err) {
        setError(err.message);
      } finally {
        setLoading(false);
      }
    };

    fetchCard();
  }, [cardId]);

  if (!cardId || (!loading && !card)) {
    return null;
  }

  if (loading) {
    return (
      <div className="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center z-50">
        <div className="bg-white rounded-2xl p-8">
          <div className="animate-spin rounded-full h-12 w-12 border-b-2 border-pink-500 mx-auto"></div>
          <p className="mt-4 text-gray-600">Loading card...</p>
        </div>
      </div>
    );
  }

  if (error) {
    return (

```

```
<div className="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center z-50 p-4">
  <div className="bg-white rounded-2xl p-8 max-w-md">
    <h2 className="text-2xl font-bold text-red-600 mb-4">Card Not Found</h2>
    <p className="text-gray-600 mb-6">
      The card you're looking for doesn't exist or has been removed.
    </p>
    <Link
      to="/"
      className="block w-full py-3 bg-pink-500 text-white rounded-lg hover:bg-pink-600 transition
text-center"
    >
      Back to All Cards
    </Link>
  </div>
</div>
);
}

return (
  <div className="fixed inset-0 bg-black bg-opacity-50 flex items-center justify-center z-50 p-4
overflow-y-auto">
  <div className="bg-white rounded-2xl max-w-2xl w-full my-8 shadow-2xl">
    <div className="relative">
      <img
        src={card.image_url}
        alt={card.title}
        className="w-full h-64 sm:h-96 object-cover rounded-t-2xl"
      />
      <Link
        to="/"
        className="absolute top-4 left-4 p-2 bg-white rounded-full shadow-lg hover:bg-gray-100
transition"
      >
        <ArrowLeft size={24} className="text-gray-700" />
      </Link>
    </div>

    <div className="p-6 sm:p-8">
      <div className="flex items-start justify-between mb-4">
        <h1 className="text-3xl font-bold text-gray-900">{card.title}</h1>
        {card.category && (
          <span className="px-3 py-1 bg-pink-100 text-pink-600 rounded-full text-sm">
            {card.category}
          </span>
        )}
      </div>

      <p className="text-gray-700 text-lg leading-relaxed mb-6 whitespace-pre-wrap">
        {card.description}
      </p>

      {((card.music_url || card.music_file_url) && (
        <AudioPlayer
          musicUrl={card.music_url}
          musicFileUrl={card.music_file_url}
        </AudioPlayer>
      ))}
    </div>
  </div>
</div>
```

```
    title={card.title}
  />
)}
```



```
<div className="mt-6 pt-6 border-t border-gray-200 text-sm text-gray-500">
  Card ID: <span className="font-mono font-semibold">{card.card_id}</span>
</div>
</div>
</div>
);
}
```