



Hola buenas noches, tengo una web con react que muestra unas tarjetas partiendo de un fichero json con datos, viene siendo una landing page, quisiera que al hacer click sobre una de ellas se muestre otra sección con más datos relacionados usando react router solamente para esta opcion con la tarjeta previamente seleccionada, me muestras un code base, gracias!

¡Hola! Para lograr lo que describes—mostrar una landing page con tarjetas y, al hacer clic en una de ellas, mostrar una sección con más datos relacionados usando **React Router**—puedes seguir una estructura como la siguiente. Aquí tienes un **code base** sencillo y claro, usando solo React y React Router (v6 o superior):

1. Estructura de archivos sugerida

- App.js
- data.json
- components/CardList.js
- components/CardDetail.js

2. Ejemplo de fichero data.json

```
[
  {
    "id": "1",
    "title": "Tarjeta 1",
    "description": "Descripción corta de la tarjeta 1",
    "details": "Información detallada sobre la tarjeta 1"
  },
  {
    "id": "2",
    "title": "Tarjeta 2",
    "description": "Descripción corta de la tarjeta 2",
    "details": "Información detallada sobre la tarjeta 2"
  }
]
```

3. Código base en React

App.js

```
import React from 'react';
import { BrowserRouter, Routes, Route } from 'react-router-dom';
import CardList from './components/CardList';
import CardDetail from './components/CardDetail';

function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<CardList />} />
        <Route path="/card/:id" element={<CardDetail />} />
      </Routes>
    </BrowserRouter>
  );
}

export default App;
```

components/CardList.js

```
import React from 'react';
import { Link } from 'react-router-dom';
import data from '../data.json';

const CardList = () => (
  <div>
    <h1>Landing Page</h1>
    <div style={{ display: 'flex', gap: '1rem' }}>
      {data.map(card => (
        <div key={card.id} style={{ border: '1px solid #ccc', padding: '1rem' }}>
          <h2>{card.title}</h2>
          <p>{card.description}</p>
          <Link to={` /card/${card.id}`}>Ver más</Link>
        </div>
      ))}
    </div>
  </div>
);

export default CardList;
```

components/CardDetail.js

```
import React from 'react';
import { useParams, Link } from 'react-router-dom';
import data from '../data.json';
```

```
const CardDetail = () => {
  const { id } = useParams();
  const card = data.find(item => item.id === id);

  if (!card) {
    return <div>Tarjeta no encontrada</div>;
  }

  return (
    <div>
      <h1>{card.title}</h1>
      <p>{card.details}</p>
      <Link to="/">Volver</Link>
    </div>
  );
};

export default CardDetail;
```

Puntos clave:

- **React Router** permite navegar entre rutas sin recargar la página, usando el componente `Link` en vez de `` para una experiencia fluida^[1] ^[2].
- La ruta `/` muestra todas las tarjetas (landing).
- La ruta `/card/:id` muestra los detalles de la tarjeta seleccionada.
- Los datos se cargan desde un fichero JSON local.

Este patrón es muy común para landings y catálogos en React, y puedes ampliarlo fácilmente. Si necesitas que la sección de detalles se muestre como un modal en la misma página (sin cambiar de ruta), el patrón sería distinto, pero usando rutas dinámicas como aquí, el usuario puede compartir el enlace directo a una tarjeta específica^[1] ^[3].

✱

1. <https://10code.es/react-router-dom/>
2. https://www.youtube.com/watch?v=WATv_Md3y_o
3. <https://hackernoon.com/es/advanced-react-router-v6-techniques-yendo-mas-alla-de-lo-basico>