



MODULE 3: SHOPPING CART FUNCTIONALITY

LESSON 8: Building the Cart Page

Learning Objective

By the end of this lesson, you will create a fully functional cart page where users can review and manage their items.

8.1 Creating the Cart Page

Step 1: Create `app/cart/page.js`

```
javascript
```

```
"use client";
```

```
import Link from 'next/link';
```

```

import { useCart } from '@app/context/CartContext';
import Header from '@app/components/Header';
import Footer from '@app/components/Footer';

export default function CartPage() {
  const { cart, removeFromCart, updateQuantity, cartTotal, clearCart } = useCart();

  if (cart.length === 0) {
    return (
      <div>
        <Header />
        <main style={{ padding: '4rem 2rem', textAlign: 'center', minHeight: '60vh' }}>
          <div style={{ maxWidth: '500px', margin: '0 auto' }}>
            <h1 style={{ fontSize: '2rem', marginBottom: '1rem' }}>Your Cart is Empty</h1>
            <p style={{ color: '#64748b', marginBottom: '2rem' }}>
              Looks like you haven't added any products yet.
            </p>
            <Link
              href="/products"
              style={{
                backgroundColor: '#1e40af',
                color: 'white',
                padding: '0.75rem 2rem',
                textDecoration: 'none',
                borderRadius: '4px',
                display: 'inline-block'
              }}
            >
              Continue Shopping
            </Link>
          </div>
        </main>
        <Footer />
      </div>
    );
  }
}

```

```
return (  
  <div>  
    <Header />  
  
    <main style={{ padding: '2rem', maxWidth: '1200px', margin: '0 auto' }}>  
      <h1 style={{ fontSize: '2rem', marginBottom: '2rem' }}>Shopping Cart</h1>  
  
      <div style={{  
        display: 'grid',  
        gridTemplateColumns: '2fr 1fr',  
        gap: '2rem'  
      }}>  
        { /* Cart Items */}  
        <div>  
          {cart.map(item => (  
            <div  
              key={item.id}  
              style={{  
                display: 'grid',  
                gridTemplateColumns: '100px 1fr auto',  
                gap: '1rem',  
                padding: '1rem',  
                borderBottom: '1px solid #e2e8f0',  
                alignItems: 'center'  
              }}  
            >  
              { /* Product Image */}  
              <div style={{  
                width: '100px',  
                height: '100px',  
                backgroundColor: '#f1f5f9',  
                borderRadius: '4px',  
                display: 'flex',  
                alignItems: 'center',  
                justifyContent: 'center',  
                color: '#64748b'  
              }}>
```



```
</div>
```

```
{/* Product Details */}
```

```
<div>
```

```
<Link
```

```
  href={`~/product/${item.id}`}
```

```
  style={{
```

```
    fontWeight: 'bold',
```

```
    color: '#1e293b',
```

```
    textDecoration: 'none',
```

```
    fontSize: '1.1rem'
```

```
  }}>
```

```
>
```

```
  {item.name}
```

```
</Link>
```

```
<p style={{ color: '#64748b', fontSize: '0.9rem', margin: '0.25rem 0' }}>
```

```
  {item.description.substring(0, 60)}...
```

```
</p>
```

```
<p style={{ fontWeight: 'bold', color: '#1e40af' }}>
```

```
  Rs. {item.price.toLocaleString()} each
```

```
</p>
```

```
</div>
```

```
{/* Quantity and Remove */}
```

```
<div style={{ textAlign: 'right' }}>
```

```
<div style={{ marginBottom: '0.5rem' }}>
```

```
<label style={{ marginRight: '0.5rem', fontSize: '0.9rem' }}>Qty:</label>
```

```
<select
```

```
  value={item.quantity}
```

```
  onChange={(e) => updateQuantity(item.id, parseInt(e.target.value))}
```

```
  style={{
```

```
    padding: '0.25rem',
```

```
    border: '1px solid #cbd5e1',
```

```
    borderRadius: '4px',
```

```
    width: '60px'
```

```
  }}>
```

```

    >
      {[1,2,3,4,5].map(num => (
        <option key={num} value={num}>{num}</option>
      ))}
    </select>
  </div>

  <p style={{ fontWeight: 'bold', marginBottom: '0.5rem' }}>
    Rs. {(item.price * item.quantity).toLocaleString()}
  </p>

  <button
    onClick={() => removeFromCart(item.id)}
    style={{
      color: '#dc2626',
      border: 'none',
      background: 'none',
      cursor: 'pointer',
      fontSize: '0.9rem'
    }}
  >
    Remove
  </button>
</div>
</div>
)))}

```

```

{ /* Clear Cart Button */ }
<div style={{ marginTop: '1rem', textAlign: 'right' }}>
  <button
    onClick={clearCart}
    style={{
      color: '#64748b',
      border: '1px solid #cbd5e1',
      background: 'white',
      padding: '0.5rem 1rem',
      borderRadius: '4px',

```

```

        cursor: 'pointer'
      }}
    >
    Clear Cart
  </button>
</div>
</div>

{/* Order Summary */}
<div style={{
  backgroundColor: '#f8fafc',
  padding: '1.5rem',
  borderRadius: '8px',
  height: 'fit-content'
}}>
  <h2 style={{ fontSize: '1.3rem', marginBottom: '1rem' }}>Order Summary</h2>

  <div style={{ marginBottom: '1rem' }}>
    <div style={{
      display: 'flex',
      justifyContent: 'space-between',
      marginBottom: '0.5rem'
    }}>
      <span>Subtotal</span>
      <span>Rs. {cartTotal.toLocaleString()}</span>
    </div>

    <div style={{
      display: 'flex',
      justifyContent: 'space-between',
      marginBottom: '0.5rem'
    }}>
      <span>Shipping</span>
      <span>Rs. 250</span>
    </div>

    <div style={{

```

```
borderTop: '1px solid #e2e8f0',
margin: '1rem 0',
paddingTop: '1rem'
}}>
<div style={{
  display: 'flex',
  justifyContent: 'space-between',
  fontWeight: 'bold',
  fontSize: '1.2rem'
}}>
  <span>Total</span>
  <span>Rs. {(cartTotal + 250).toLocaleString()}</span>
</div>
</div>
</div>
```

```
<Link
href="/checkout"
style={{
  display: 'block',
  backgroundColor: '#1e40af',
  color: 'white',
  textAlign: 'center',
  padding: '1rem',
  textDecoration: 'none',
  borderRadius: '4px',
  fontWeight: 'bold',
  marginBottom: '1rem'
}}
>
```

```
  Proceed to Checkout
</Link>
```

```
<Link
href="/products"
style={{
  display: 'block',
```

```

        color: '#1e40af',
        textAlign: 'center',
        textDecoration: 'none'
      }}
    >
    Continue Shopping
  </Link>
</div>
</div>
</main>

<Footer />
</div>
);
}

```

8.2 Adding Cart Count Badge to Header

Step 1: Update `app/components/Header.js`

```

javascript
import Link from 'next/link';
import { useCart } from '@app/context/CartContext';

function Header() {
  const { cartItemsCount } = useCart();

  return (
    <header style={{
      backgroundColor: '#1e40af',
      color: 'white',
      padding: '1rem',
      display: 'flex',

```

```

justifyContent: 'space-between',
alignItems: 'center'
}}>
<Link href="/" style={{ textDecoration: 'none', color: 'white' }}>
  <div style={{ fontSize: '1.5rem', fontWeight: 'bold' }}>
    🏠 CeylonMart
  </div>
</Link>

<nav style={{ display: 'flex', alignItems: 'center', gap: '1rem' }}>
  <Link href="/" style={{ color: 'white', textDecoration: 'none' }}>
    Home
  </Link>
  <Link href="/products" style={{ color: 'white', textDecoration: 'none' }}>
    Products
  </Link>
  <Link href="/cart" style={{ color: 'white', textDecoration: 'none', position: 'relative' }}>
    Cart
    {cartItemsCount > 0 && (
      <span style={{
        position: 'absolute',
        top: '-8px',
        right: '-12px',
        backgroundColor: '#dc2626',
        color: 'white',
        borderRadius: '9999px',
        padding: '0.125rem 0.5rem',
        fontSize: '0.75rem',
        fontWeight: 'bold'
      }}>
        }}>
      {cartItemsCount}
    </span>
  )}
</Link>
</nav>
</header>
);

```

```
}
```

```
export default Header;
```

8.3 Adding Cart Summary Component

Create a reusable cart summary component that can be used in multiple places.

Step 1: Create `app/components/CartSummary.js`

```
javascript
```

```
import { useCart } from '@app/context/CartContext';
```

```
export default function CartSummary() {  
  const { cart, cartTotal } = useCart();
```

```
  if (cart.length === 0) {  
    return null;  
  }
```

```
  return (  
    <div style={{  
      position: 'fixed',  
      bottom: '2rem',  
      right: '2rem',  
      backgroundColor: 'white',  
      boxShadow: '0 10px 15px -3px rgba(0,0,0,0.1)',  
      borderRadius: '8px',  
      padding: '1rem',  
      width: '300px',  
      zIndex: 1000
```

```

}}>
<h3 style={{ margin: '0 0 0.5rem 0' }}>Cart Summary</h3>
<p style={{ margin: '0 0 0.5rem 0', color: '#64748b' }}>
  {cart.length} item{cart.length !== 1 ? 's': ''}
</p>
<p style={{ fontWeight: 'bold', margin: '0 0 1rem 0' }}>
  Total: Rs. {cartTotal.toLocaleString()}
</p>
<Link
  href="/cart"
  style={{
    display: 'block',
    backgroundColor: '#1e40af',
    color: 'white',
    textAlign: 'center',
    padding: '0.5rem',
    textDecoration: 'none',
    borderRadius: '4px'
  }}
  >
  View Cart
</Link>
</div>
);
}

```

Lesson 8 Summary

- Cart page displays all items with quantities
- Order summary shows totals
- Cart count badge provides quick visual feedback
- Empty cart state improves UX

- Floating cart summary helps users track total
-

Practice Exercises

1. Add quantity update buttons (+ and -) instead of dropdown
2. Implement coupon code input with discount calculation
3. Show shipping options (standard, express, free)
4. Challenge: Add "Save for Later" feature