Shopping Cart - List Items

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

List Items in the Shopping Cart

© luv2code LLC

New Component: CartDetailsComponent

© luv2code LLC

New Component: CartDetailsComponent

© luv2code LLC

New Component: CartDetailsComponent

© luv2code LLC

Development Process

Step-By-Step

© luv2code LLC

Development Process

Step-By-Step

1. Create new component: CartDetailsComponent

© luv2code LLC

Development Process

Step-By-Step

1. Create new component: CartDetailsComponent
2. Add new route for CartDetailsComponent

© luv2code LLC

Development Process

Step-By-Step

1. Create new component: CartDetailsComponent
2. Add new route for CartDetailsComponent
3. Update link for Shopping Cart icon

© luv2code LLC

Development Process

Step-By-Step

1. Create new component: CartDetailsComponent
2. Add new route for CartDetailsComponent
3. Update link for Shopping Cart icon
4. Modify CartDetailsComponent to retrieve cart items

© luv2code LLC

Development Process

Step-By-Step

1. Create new component: CartDetailsComponent
2. Add new route for CartDetailsComponent
3. Update link for Shopping Cart icon
4. Modify CartDetailsComponent to retrieve cart items
5. Add HTML template for CartDetailsComponent

© luv2code LLC

Step 1: Create new component for Cart Details

© luv2code LLC

Step 1: Create new component for Cart Details

```
> ng generate component components/cart-details
```


© luv2code LLC

Step 2: Add new route for CartDetailsComponent

© luv2code LLC

Step 2: Add new route for CartDetailsComponent

File: app.module.ts

```
const routes: Routes = [
  {path: 'cart-details', component: CartDetailsComponent},
  ...

];
```


© luv2code LLC

Step 3: Update link for Shopping Cart icon

© luv2code LLC

Step 3: Update link for Shopping Cart icon

File: cart-status.component.html

```
<div class="cart-area d-n">

    <a routerLink="/cart-details">
        <div class="total">{{ totalPrice | currency: 'USD' }}
            <span>{{ totalQuantity }}</span>
        </div>
        <i class="fa fa-shopping-cart" aria-hidden="true"></i>
    </a>

</div>
```


© luv2code LLC

Step 3: Update link for Shopping Cart icon

File: cart-status.component.html

```
<div class="cart-area d-n">

    <a routerLink="/cart-details">
        <div class="total">{{ totalPrice | currency: 'USD' }}
            <span>{{ totalQuantity }}</span>
        </div>
        <i class="fa fa-shopping-cart" aria-hidden="true"></i>
    </a>

</div>
```


© luv2code LLC

Step 3: Update link for Shopping Cart icon

File: cart-status.component.html

```
<div class="cart-area d-n">

    <a routerLink="/cart-details">
        <div class="total">{{ totalPrice | currency: 'USD' }}
            <span>{{ totalQuantity }}</span>
        </div>
        <i class="fa fa-shopping-cart" aria-hidden="true"></i>
    </a>

</div>
```


© luv2code LLC

Step 3: Update link for Shopping Cart icon

File: cart-status.component.html

```
<div class="cart-area d-n">

    <a routerLink="/cart-details">
        <div class="total">{{ totalPrice | currency: 'USD' }}
            <span>{{ totalQuantity }}</span>
        </div>
        <i class="fa fa-shopping-cart" aria-hidden="true"></i>
    </a>

</div>
```

When user clicks `CartStatusComponent` then use new route: `/cart-details` to view `CartDetailsComponent`

© luv2code LLC

Step 4: Modify CartDetailsComponent to retrieve cart items

© luv2code LLC

Step 4: Modify CartDetailsComponent to retrieve cart items

File: cart-details.component.ts

```
export class CartDetailsComponent implements OnInit {

  cartItems: CartItem[] = [];
  totalPrice: number = 0;
  totalQuantity: number = 0;

  constructor(private cartService: CartService) { }

  ngOnInit() {
    this.listCartDetails();
  }
```


© luv2code LLC

Step 4: Modify CartDetailsComponent to retrieve cart items

File: cart-details.component.ts

```
export class CartDetailsComponent implements OnInit {

  cartItems: CartItem[] = [];
  totalPrice: number = 0;
  totalQuantity: number = 0;

  constructor(private cartService: CartService) { }

  ngOnInit() {
    this.listCartDetails();
  }
```

```
  listCartDetails() {

    // get a handle to the cart items
    this.cartItems = this.cartService.cartItems;

    // subscribe to the cart totalPrice
    this.cartService.totalPrice.subscribe(
      data => this.totalPrice = data
    );

    // subscribe to the cart totalQuantity
    this.cartService.totalQuantity.subscribe(
      data => this.totalQuantity = data
    );

    // compute cart total price and total quantity
    this.cartService.computeCartTotals();
  }

}
```


© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
<table class="table table-bordered">
   <tr>
      <th width="20%">Product Image</th>
      <th width="50%">Product Detail</th>
      <th width="30%"></th>
   </tr>
```

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
<table class="table table-bordered">
   <tr>
      <th width="20%">Product Image</th>
      <th width="50%">Product Detail</th>
      <th width="30%"></th>
   </tr>
```

```
<tr *ngFor="let tempCartItem of cartItems">
```

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
<table class="table table-bordered">
   <tr>
      <th width="20%">Product Image</th>
      <th width="50%">Product Detail</th>
      <th width="30%"></th>
   </tr>
```

```
<tr *ngFor="let tempCartItem of cartItems">
  <td>
    <img src="{{ tempCartItem.imageUrl }}" class="img-responsive" width="150px" />
  </td>
```

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
<table class="table table-bordered">
   <tr>
      <th width="20%">Product Image</th>
      <th width="50%">Product Detail</th>
      <th width="30%"></th>
   </tr>
```

```
<tr *ngFor="let tempCartItem of cartItems">
  <td>
    <img src="{{ tempCartItem.imageUrl }}" class="img-responsive" width="150px" />
  </td>

  <td>
    <p>{{ tempCartItem.name }}</p>
    <p>{{ tempCartItem.unitPrice | currency:'USD' }} </p>
  </td>
```

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
<table class="table table-bordered">
   <tr>
      <th width="20%">Product Image</th>
      <th width="50%">Product Detail</th>
      <th width="30%"></th>
   </tr>
```

```
  <tr *ngFor="let tempCartItem of cartItems">
    <td>
      <img src="{{ tempCartItem.imageUrl }}" class="img-responsive" width="150px" />
    </td>

    <td>
      <p>{{ tempCartItem.name }}</p>
      <p>{{ tempCartItem.unitPrice | currency:'USD' }} </p>
    </td>

    <td>
      <div class="items">
        <label>Quantity:</label> {{ tempCartItem.quantity }}
      </div>

      <p class="mt-2">Subtotal: {{ tempCartItem.quantity * tempCartItem.unitPrice | currency: 'USD' }}</p>
    </td>
  </tr>

</table>
```

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
...
 <tr>
  <td colspan="2"></td>
  <td style="font-weight: bold">

   <p>Total Quantity: {{ totalQuantity }}</p>

   <p>Shipping: FREE</p>

   <p>Total Price: {{ totalPrice | currency: 'USD' }}</p>

  </td>
 </tr>
```


© luv2code LLC

Step 5: Add HTML template for CartDetailsComponent

File: cart-details.component.ts

```
...
 <tr>
  <td colspan="2"></td>
  <td style="font-weight: bold">

   <p>Total Quantity: {{ totalQuantity }}</p>

   <p>Shipping: FREE</p>

   <p>Total Price: {{ totalPrice | currency: 'USD' }}</p>

  </td>
 </tr>
```

Total Quantity: 2

Shipping: FREE

Total Price: $37.98

© luv2code LLC