SHOPMART

A Shopping App

-DEEPIKA G

Problem Statement

- ✓ Managing dynamic UI components efficiently.
- ✓ Handling client-side routing seamlessly.
- ✓ Ensuring smooth state management with real-time data updates.
- ✓ Implementing reusable and maintainable component-based architecture.
- ✓ Optimizing API handling and asynchronous data flow.
- ✓ Enhancing UI behavior using directives.
- ✓ Improving scalability and performance of web applications.

Project Overview

Project Name:**SHOPMART**

- ✓ Developed using **Angular** for a scalable and maintainable web application.
- ✓ Implements **component-based architecture** for modular development.
- ✓ Utilizes **client-side routing** for seamless navigation.
- ✓ Leverages **RxJS Observables** for efficient state management and API handling.
- ✓ Enhances UI with **Structural & Attribute Directives**.
- ✓ Ensures responsive and optimized performance with Angular best practices.

Login page

Allows users to authenticate into the system.

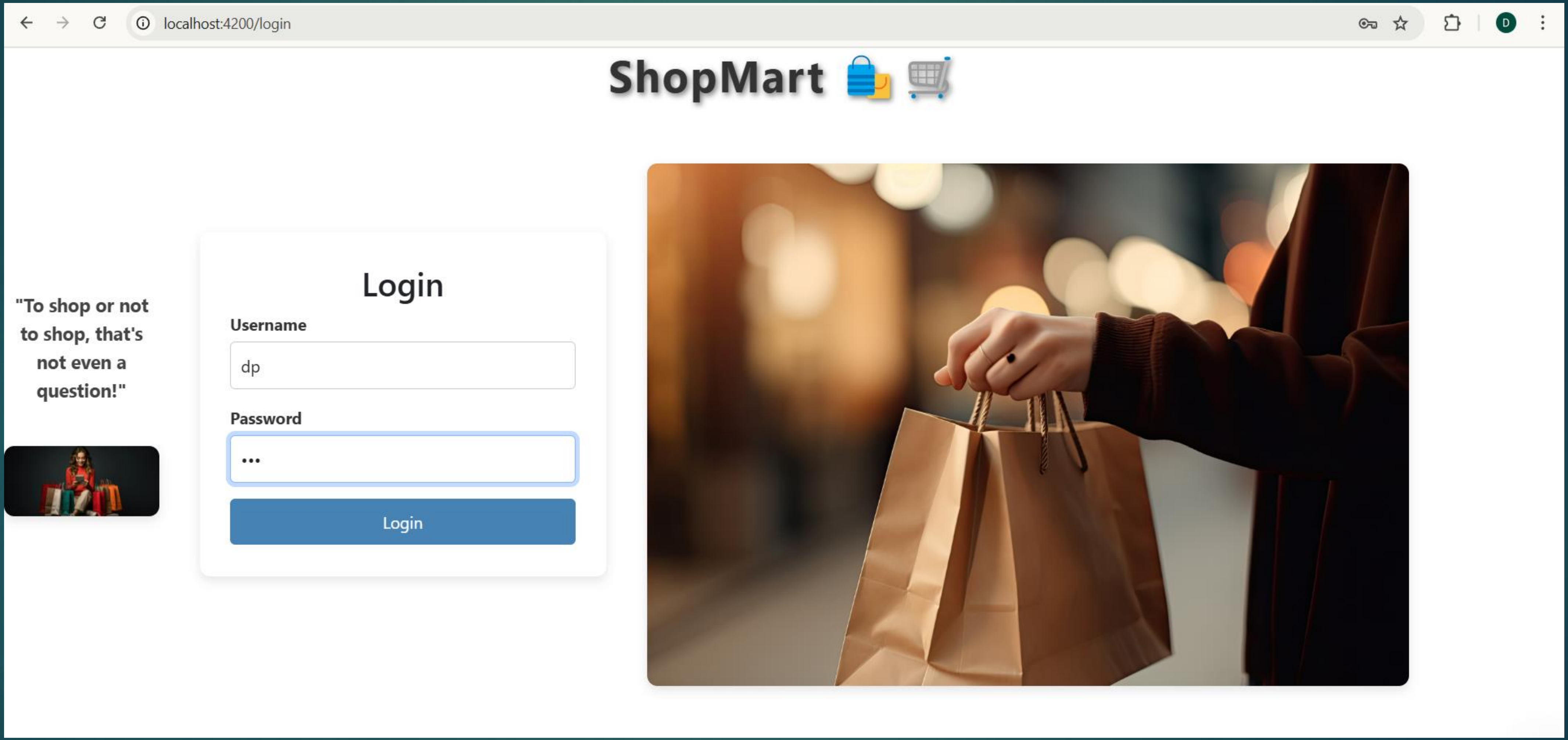
Features

- ✓ Template-Driven Form – Handles user authentication seamlessly
- ✓ Custom Pipe – Formats user input (e.g., converts email to lowercase)
- ✓ Local Storage – Stores login credentials for session management
- ✓ Routing Guard – Restricts access to authenticated users

Functionality

- ✓ Users enter credentials and log in securely
- ✓ Form validation implemented using Angular Validators
- ✓ Uses @Input() to receive dynamic form configurations
- ✓ Uses @Output() to emit login status to the parent component

Login Page



localhost:4200/login

ShopMart 🛍️ 🛒

"To shop or not to shop, that's not even a question!"

Login

Username

dp

Password

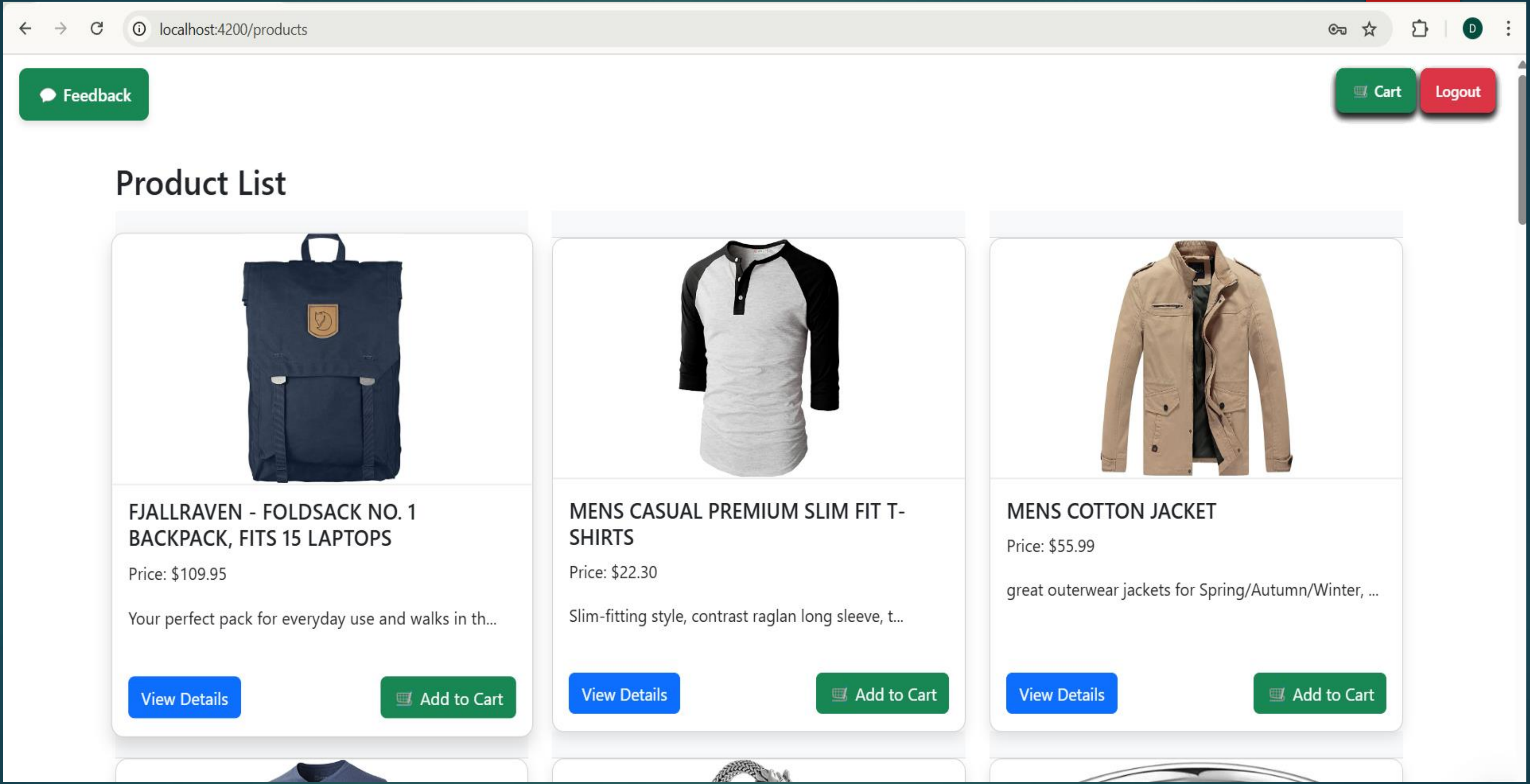
•••

Products page

Displays a list of products fetched from an API.

Features & Concepts Implemented

- API Calls: Fetches product details from an API.
- **Routing & Parameters:**
- ✓ Clicking a product dynamically loads its details using routerLink.
- ✓ ActivatedRoute extracts product ID from the URL.
- **Components**:
- ✓ Parent component displays all products.
- ✓ Child component manages individual product cards.
- **Pipes**:
- ✓ Inbuilt Pipes: CurrencyPipe formats product prices.
- ✓ Custom pipes:Formats the length of the product description by concatenating certain length with …



Product List

FJALLRAVEN - FOLDSACK NO. 1 BACKPACK, FITS 15 LAPTOPS

Price: $109.95

Your perfect pack for everyday use and walks in th...

MENS CASUAL PREMIUM SLIM FIT T-SHIRTS

Price: $22.30

Slim-fitting style, contrast raglan long sleeve, t...

MENS COTTON JACKET

Price: $55.99

great outerwear jackets for Spring/Autumn/Winter, ...

Product Details Page

Displays complete product information, including name, price, description, images.

Features & Concepts Implemented

- Routing Parameters (ActivatedRoute) – Extracts product ID from the URL to fetch details dynamically.
- API Calls (HttpClient) – Retrieves product data from an API.
- **Parent-Child Communication** (@Input, @Output) –

✓ Parent Component: Fetches and manages product details.

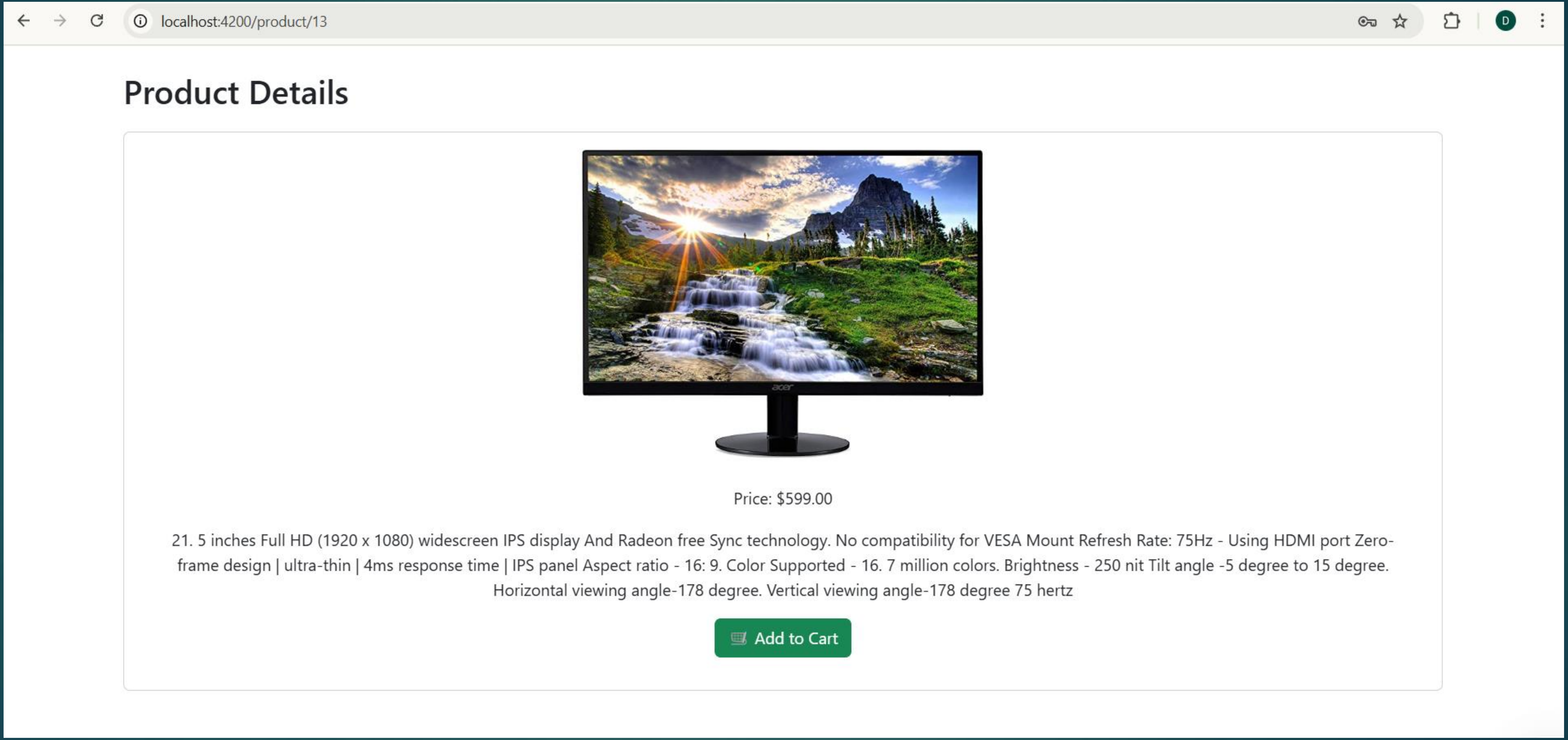
✓ Child Component: Displays images, pricing, and specifications.

Add to Cart Functionality:

✓ Uses @Output() to emit events for cart updates.

✓ Stores cart data in local storage for persistence.

Product Details Page



Product Details

Price: $599.00

21. 5 inches Full HD (1920 x 1080) widescreen IPS display And Radeon free Sync technology. No compatibility for VESA Mount Refresh Rate: 75Hz - Using HDMI port Zero-frame design | ultra-thin | 4ms response time | IPS panel Aspect ratio - 16: 9. Color Supported - 16. 7 million colors. Brightness - 250 nit Tilt angle -5 degree to 15 degree. Horizontal viewing angle-178 degree. Vertical viewing angle-178 degree 75 hertz

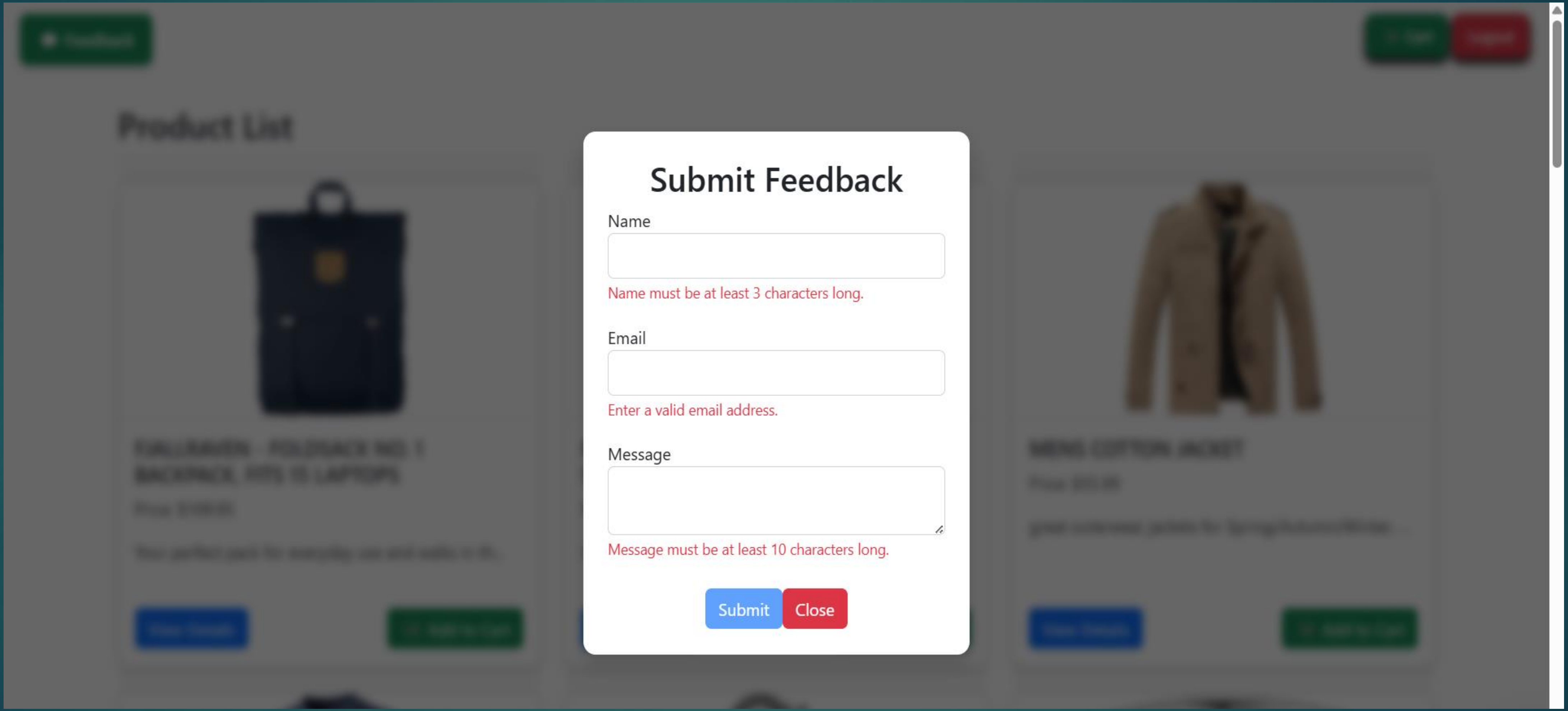
Cart Page

Manages products added to the shopping cart.

Features & Concepts Implemented

- **Parent-Child Communication:**
  - ✓ **Parent Component:** Cart stores products.
  - ✓ **Child Component:** Handles product quantity updates.
- **Local Storage:** Persists cart data even after page refresh.
- **Event Binding:**
  - ✓ Listens to quantity updates and emits data to the parent component.
- **Routing Guard (AuthGuard):** Restricts access to logged-in users.

Feedback form with validation



Submit Feedback

Name

Name must be at least 3 characters long.

Email

Enter a valid email address.

Message

Message must be at least 10 characters long.

Feedback form

The Feedback Page allows users to submit reviews about their shopping experience.

Features

- ✓ Reactive Forms (Model-Driven Forms) – Handles structured user feedback.
- ✓ Two-Way Binding ([(ngModel)]) – Updates form inputs in real time.
- ✓ Custom Validator – Ensures feedback length meets minimum criteria.
- ✓ Observables – Stores feedback asynchronously.

Functionality

- ✓ Captures user reviews and comments.
- ✓ Ensures real-time validation of input fields.
- ✓ Stores feedback for further analysis in the api(feedback.json)

Logout Page

Logs out users and clears session data.

Features & Concepts Implemented

- **Local Storage Management:** Clears user session data on logout.
- **Routing Navigation:** Redirects users to the login page post-logout.
- **Routing Guard (AuthGuard):** Prevents logged-out users from accessing restricted pages.