

Daywise_Assignment_Javascript_PMS_App

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Product Management System</title>

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css">

  <style>

    .card-img-top {

      height: 200px;

      object-fit: cover;

    }

    .product-card {

      transition: transform 0.3s;

    }

    .product-card:hover {

      transform: translateY(-5px);

      box-shadow: 0 10px 20px rgba(0,0,0,0.1);

    }

    .cart-badge {

      position: absolute;

      top: -5px;

      right: -5px;
```

```

    }

    .hero-section {
        background: linear-gradient(rgba(0,0,0,0.6), rgba(0,0,0,0.6)),
url('https://images.unsplash.com/photo-1552664730-d307ca884978?ixlib=rb-
4.0.3&auto=format&fit=crop&w=1200&q=80');

        background-size: cover;

        background-position: center;

        color: white;

        padding: 100px 0;

        margin-bottom: 30px;
    }

    .feature-icon {
        font-size: 2rem;

        margin-bottom: 1rem;

        color: #0d6efd;
    }

    .dashboard-card {
        border: none;

        border-radius: 10px;

        box-shadow: 0 5px 15px rgba(0,0,0,0.08);
    }
</style>
</head>
<body>

    <!-- Navigation will be dynamically loaded here -->

    <div id="navbar-container"></div>


    <!-- Main content area where pages will be loaded -->

    <div class="container mt-4" id="main-content">

```

```

        <!-- Welcome page will be loaded here by default -->

</div>


<!-- Footer -->

<footer class="bg-dark text-white text-center py-4 mt-5">

    <div class="container">

        <p>&copy; 2023 Product Management System. All rights reserved.</p>

    </div>

</footer>


<!-- Scripts -->

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></scrip
t>

<script>

    // ===== DATA MANAGEMENT =====

    // Initialize data in localStorage if not exists

    if (!localStorage.getItem('users')) {

        const initialUsers = [

            { id: 1, name: "Admin User", email: "admin@pms.com", password: "admin123",
role: "admin" },

            { id: 2, name: "John Doe", email: "john@example.com", password: "user123", role:
"user" }

        ];

        localStorage.setItem('users', JSON.stringify(initialUsers));

    }


    if (!localStorage.getItem('products')) {

        const initialProducts = [

```

```

    { id: 1, name: "Wireless Headphones", description: "High-quality wireless headphones with noise cancellation", price: 129.99, image: "https://images.unsplash.com/photo-1505740420928-5e560c06d30e?ixlib=rb-4.0.3&auto=format&fit=crop&w=600&q=80" },

    { id: 2, name: "Smartphone", description: "Latest smartphone with advanced camera system", price: 699.99, image: "https://images.unsplash.com/photo-1511707171634-5f897ff02aa9?ixlib=rb-4.0.3&auto=format&fit=crop&w=600&q=80" },

    { id: 3, name: "Laptop", description: "Powerful laptop for work and entertainment", price: 999.99, image: "https://images.unsplash.com/photo-1496181133206-80ce9b88a853?ixlib=rb-4.0.3&auto=format&fit=crop&w=600&q=80" },

    { id: 4, name: "Smart Watch", description: "Feature-rich smartwatch with health monitoring", price: 199.99, image: "https://images.unsplash.com/photo-1523275335684-37898b6baf30?ixlib=rb-4.0.3&auto=format&fit=crop&w=600&q=80" }

];

localStorage.setItem('products', JSON.stringify(initialProducts));
}

// Get current user from localStorage
function getCurrentUser() {
    return JSON.parse(localStorage.getItem('currentUser')) || null;
}

// Set current user in localStorage
function setCurrentUser(user) {
    if (user) {
        localStorage.setItem('currentUser', JSON.stringify(user));
    } else {
        localStorage.removeItem('currentUser');
    }
}

```

```
// Get all users  
function getUsers() {  
    return JSON.parse(localStorage.getItem('users')) || [];  
}
```

```
// Add new user  
function addUser(user) {  
    const users = getUsers();  
    user.id = users.length > 0 ? Math.max(...users.map(u => u.id)) + 1 : 1;  
    users.push(user);  
    localStorage.setItem('users', JSON.stringify(users));  
}
```

```
// Get all products  
function getProducts() {  
    return JSON.parse(localStorage.getItem('products')) || [];  
}
```

```
// Add new product  
function addProduct(product) {  
    const products = getProducts();  
    product.id = products.length > 0 ? Math.max(...products.map(p => p.id)) + 1 : 1;  
    products.push(product);  
    localStorage.setItem('products', JSON.stringify(products));  
}
```

```
// Update product  
function updateProduct(updatedProduct) {
```

```

const products = getProducts();

const index = products.findIndex(p => p.id === updatedProduct.id);

if (index !== -1) {
    products[index] = updatedProduct;

    localStorage.setItem('products', JSON.stringify(products));
}
}

// Delete product
function deleteProduct(productId) {
    let products = getProducts();

    products = products.filter(p => p.id !== productId);

    localStorage.setItem('products', JSON.stringify(products));
}

// Get user cart
function getCart() {
    const user = getCurrentUser();

    if (!user) return [];

    const cart = JSON.parse(localStorage.getItem(`cart_${user.id}`)) || [];

    return cart;
}

// Add to cart
function addToCart(product) {
    const user = getCurrentUser();

    if (!user) {

```

```
    alert('Please login to add products to cart');  
    loadPage('login');  
    return;  
}
```

```
const cart = getCart();  
const existingItem = cart.find(item => item.productId === product.id);
```

```
if (existingItem) {  
    existingItem.quantity += 1;  
} else {  
    cart.push({  
        productId: product.id,  
        name: product.name,  
        price: product.price,  
        image: product.image,  
        quantity: 1  
    });  
}
```

```
localStorage.setItem(`cart_${user.id}`, JSON.stringify(cart));  
updateCartBadge();  
alert('Product added to cart!');  
}
```

```
// Remove from cart  
function removeFromCart(productId) {  
    const user = getCurrentUser();
```

```
if (!user) return;

let cart = getCart();
cart = cart.filter(item => item.productId !== productId);

localStorage.setItem(`cart_${user.id}`, JSON.stringify(cart));
updateCartBadge();
loadPage('cart'); // Reload cart page
}

// Clear cart
function clearCart() {
  const user = getCurrentUser();
  if (!user) return;

  localStorage.setItem(`cart_${user.id}`, JSON.stringify([]));
  updateCartBadge();
}

// Get cart item count
function getCartItemCount() {
  const cart = getCart();
  return cart.reduce((total, item) => total + item.quantity, 0);
}

// Update cart badge in navbar
function updateCartBadge() {
  const cartBadge = document.getElementById('cart-badge');
```



```

if (cartBadge) {
    const count = getCartItemCount();
    cartBadge.textContent = count;
    cartBadge.style.display = count > 0 ? 'inline-block' : 'none';
}
}

```

```

// ===== NAVIGATION =====

```

```

// Load navigation based on user role

```

```

function loadNavigation() {

```

```

    const user = getCurrentUser();

```

```

    const navbarContainer = document.getElementById('navbar-container');

```

```

    let navbarHtml = `

```

```

        <nav class="navbar navbar-expand-lg navbar-dark bg-primary">

```

```

            <div class="container">

```

```

                <a class="navbar-brand" href="#" onclick="loadPage('welcome')">

```

```

                    <i class="fas fa-boxes me-2"></i>PMS

```

```

                </a>

```

```

                <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-
bs-target="#navbarNav">

```

```

                    <span class="navbar-toggler-icon"></span>

```

```

                </button>

```

```

                <div class="collapse navbar-collapse" id="navbarNav">

```

```

                    <ul class="navbar-nav me-auto">

```

```

        `;

```

```

    if (user) {

```

```
if (user.role === 'admin') {  
  navbarHtml += `  
    <li class="nav-item">  
      <a class="nav-link" href="#" onclick="loadPage('admin-dashboard')">  
        <i class="fas fa-tachometer-alt me-1"></i>Dashboard  
      </a>  
    </li>  
    <li class="nav-item">  
      <a class="nav-link" href="#" onclick="loadPage('add-product')">  
        <i class="fas fa-plus-circle me-1"></i>Add Product  
      </a>  
    </li>  
    <li class="nav-item">  
      <a class="nav-link" href="#" onclick="loadPage('product-list')">  
        <i class="fas fa-list me-1"></i>Product List  
      </a>  
    </li>  
  `;  
} else {  
  navbarHtml += `  
    <li class="nav-item">  
      <a class="nav-link" href="#" onclick="loadPage('user-dashboard')">  
        <i class="fas fa-tachometer-alt me-1"></i>Dashboard  
      </a>  
    </li>  
    <li class="nav-item">  
      <a class="nav-link" href="#" onclick="loadPage('product-list')">  
        <i class="fas fa-shopping-bag me-1"></i>Products
```

```

        </a>
    </li>
    <li class="nav-item position-relative">
        <a class="nav-link" href="#" onclick="loadPage('cart')">
            <i class="fas fa-shopping-cart me-1"></i>Cart
            <span class="position-absolute top-0 start-100 translate-middle badge
rounded-pill bg-danger" id="cart-badge"></span>
        </a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#" onclick="loadPage('checkout')">
            <i class="fas fa-cash-register me-1"></i>Checkout
        </a>
    </li>
`;
}

```

navbarHtml += `

```

</ul>
<ul class="navbar-nav">
    <li class="nav-item">
        <span class="nav-link">Welcome, ${user.name}</span>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#" onclick="logout()">
            <i class="fas fa-sign-out-alt me-1"></i>Logout
        </a>
    </li>

```

```

        </ul>
    `;
} else {
    navbarHtml += `
        </ul>
        <ul class="navbar-nav">
            <li class="nav-item">
                <a class="nav-link" href="#" onclick="loadPage('login')">
                    <i class="fas fa-sign-in-alt me-1"></i>Login
                </a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#" onclick="loadPage('signup')">
                    <i class="fas fa-user-plus me-1"></i>Sign Up
                </a>
            </li>
        </ul>
    `;
}

```

```

    navbarHtml += `
        </div>
    </div>
</nav>
`;

```

```

navbarContainer.innerHTML = navbarHtml;
updateCartBadge();

```

```
}
```

```
// ===== PAGE LOADING =====
```

```
// Load a specific page
```

```
function loadPage(page) {
```

```
    const mainContent = document.getElementById('main-content');
```

```
    switch(page) {
```

```
        case 'welcome':
```

```
            loadWelcomePage(mainContent);
```

```
            break;
```

```
        case 'signup':
```

```
            loadSignupPage(mainContent);
```

```
            break;
```

```
        case 'login':
```

```
            loadLoginPage(mainContent);
```

```
            break;
```

```
        case 'admin-dashboard':
```

```
            loadAdminDashboard(mainContent);
```

```
            break;
```

```
        case 'user-dashboard':
```

```
            loadUserDashboard(mainContent);
```

```
            break;
```

```
        case 'add-product':
```

```
            loadAddProductPage(mainContent);
```

```
            break;
```

```
        case 'product-list':
```

```
            loadProductListPage(mainContent);
```

```

        break;
    case 'cart':
        loadCartPage(mainContent);
        break;
    case 'checkout':
        loadCheckoutPage(mainContent);
        break;
    default:
        loadWelcomePage(mainContent);
}

// Update navigation after page load
loadNavigation();
}

// Load welcome page
function loadWelcomePage(container) {
    container.innerHTML = `
        <div class="hero-section text-center rounded">
            <h1 class="display-4 fw-bold">Product Management System</h1>
            <p class="lead">Efficiently manage products and streamline your shopping
experience</p>
            <div class="mt-4">
                <button class="btn btn-primary btn-lg me-2"
onclick="loadPage('login')">Login</button>
                <button class="btn btn-outline-light btn-lg" onclick="loadPage('signup')">Sign
Up</button>
            </div>
        </div>
    `;
}

```

```
<div class="row mt-5">

  <div class="col-md-4 mb-4">

    <div class="card dashboard-card h-100 text-center">

      <div class="card-body">

        <div class="feature-icon">

          <i class="fas fa-cog"></i>

        </div>

        <h3 class="card-title">Admin Features</h3>

        <p class="card-text">Manage your product catalog with full CRUD
operations. Add, edit, or remove products with ease.</p>

      </div>

    </div>

  </div>

  <div class="col-md-4 mb-4">

    <div class="card dashboard-card h-100 text-center">

      <div class="card-body">

        <div class="feature-icon">

          <i class="fas fa-shopping-cart"></i>

        </div>

        <h3 class="card-title">User Features</h3>

        <p class="card-text">Browse products, add them to your cart, and
complete your purchase seamlessly.</p>

      </div>

    </div>

  </div>

  <div class="col-md-4 mb-4">

    <div class="card dashboard-card h-100 text-center">

      <div class="card-body">
```

```

        <div class="feature-icon">
            <i class="fas fa-lock"></i>
        </div>

        <h3 class="card-title">Secure Access</h3>

        <p class="card-text">Role-based authentication ensures that users only
access appropriate features.</p>

    </div>

</div>

</div>

</div>
`;
}

```

```
// Load signup page
```

```

function loadSignupPage(container) {
    container.innerHTML = `
        <div class="row justify-content-center">
            <div class="col-md-6">
                <div class="card dashboard-card">
                    <div class="card-header bg-primary text-white">
                        <h3 class="card-title mb-0">Create Account</h3>
                    </div>
                    <div class="card-body">
                        <form id="signup-form">
                            <div class="mb-3">
                                <label for="name" class="form-label">Full Name</label>
                                <input type="text" class="form-control" id="name" required>
                            </div>

```



```

<div class="mb-3">
    <label for="email" class="form-label">Email address</label>
    <input type="email" class="form-control" id="email" required>
</div>

<div class="mb-3">
    <label for="password" class="form-label">Password</label>
    <input type="password" class="form-control" id="password"
required>

</div>

<div class="mb-3">
    <label for="role" class="form-label">Role</label>
    <select class="form-select" id="role" required>
        <option value="user">User</option>
        <option value="admin">Admin</option>
    </select>
</div>

<button type="submit" class="btn btn-primary w-100">Sign
Up</button>

</form>

<div class="text-center mt-3">
    <p>Already have an account? <a href="#"
onclick="loadPage('login')">Login here</a></p>
</div>

</div>

</div>

</div>

</div>

</div>
;

```

```

document.getElementById('signup-form').addEventListener('submit', function(e) {
    e.preventDefault();

    const name = document.getElementById('name').value;
    const email = document.getElementById('email').value;
    const password = document.getElementById('password').value;
    const role = document.getElementById('role').value;

    // Check if email already exists
    const users = getUsers();
    if (users.find(user => user.email === email)) {
        alert('Email already registered. Please use a different email.');
```

return;

```
    }

    // Add new user
    addUser({ name, email, password, role });

    alert('Account created successfully! Please login.');
```

loadPage('login');

```
});
}

// Load login page
function loadLoginPage(container) {
    container.innerHTML = `
        <div class="row justify-content-center">
            <div class="col-md-6">
```

```

<div class="card dashboard-card">
  <div class="card-header bg-primary text-white">
    <h3 class="card-title mb-0">Login to Your Account</h3>
  </div>
  <div class="card-body">
    <form id="login-form">
      <div class="mb-3">
        <label for="login-email" class="form-label">Email address</label>
        <input type="email" class="form-control" id="login-email" required>
      </div>
      <div class="mb-3">
        <label for="login-password" class="form-label">Password</label>
        <input type="password" class="form-control" id="login-password"
required>
      </div>
      <button type="submit" class="btn btn-primary w-100">Login</button>
    </form>
    <div class="text-center mt-3">
      <p>Don't have an account? <a href="#"
onclick="loadPage('signup')">Sign up here</a></p>
    </div>
  </div>
</div>
</div>
</div>
</div>
;

```

```

document.getElementById('login-form').addEventListener('submit', function(e) {
  e.preventDefault();

```

```
const email = document.getElementById('login-email').value;
const password = document.getElementById('login-password').value;

// Find user with matching credentials
const users = getUsers();
const user = users.find(u => u.email === email && u.password === password);

if (user) {
  setCurrentUser(user);

  if (user.role === 'admin') {
    loadPage('admin-dashboard');
  } else {
    loadPage('user-dashboard');
  }
} else {
  alert('Invalid email or password. Please try again.');
```

```
});
}

// Load admin dashboard
function loadAdminDashboard(container) {
  const products = getProducts();

  container.innerHTML = `
    <div class="d-flex justify-content-between align-items-center mb-4">
```

```

    <h2>Admin Dashboard</h2>

    <span class="badge bg-primary">${products.length} Products</span>

</div>

<div class="row">

    <div class="col-md-4 mb-4">

        <div class="card dashboard-card text-center">

            <div class="card-body">

                <div class="feature-icon text-success">

                    <i class="fas fa-plus-circle"></i>

                </div>

                <h5 class="card-title">Add Product</h5>

                <p class="card-text">Add a new product to your catalog</p>

                <button class="btn btn-success" onclick="loadPage('add-product')">Add
Product</button>

            </div>

        </div>

    </div>

    <div class="col-md-4 mb-4">

        <div class="card dashboard-card text-center">

            <div class="card-body">

                <div class="feature-icon text-info">

                    <i class="fas fa-list"></i>

                </div>

                <h5 class="card-title">Manage Products</h5>

                <p class="card-text">View, edit or delete products</p>

                <button class="btn btn-info text-white" onclick="loadPage('product-
list')">Product List</button>

```

</div>

</div>

</div>

<div class="col-md-4 mb-4">

<div class="card dashboard-card text-center">

<div class="card-body">

<div class="feature-icon text-warning">

<i class="fas fa-chart-bar"></i>

</div>

<h5 class="card-title">Reports</h5>

<p class="card-text">View sales and inventory reports</p>

<button class="btn btn-warning" disabled>Coming Soon</button>

</div>

</div>

</div>

</div>

<div class="card dashboard-card mt-4">

<div class="card-header bg-primary text-white">

<h5 class="mb-0">Recent Products</h5>

</div>

<div class="card-body">

`\${products.length > 0 ? `

<div class="table-responsive">

<table class="table table-hover">

<thead>

<tr>

```

        <th>Name</th>

        <th>Price</th>

        <th>Actions</th>

    </tr>

</thead>

<tbody>

    ${products.slice(0, 5).map(product => `

        <tr>

            <td>${product.name}</td>

            <td>${product.price.toFixed(2)}</td>

            <td>

                <button class="btn btn-sm btn-outline-primary"
onclick="editProduct(${product.id})">Edit</button>

                <button class="btn btn-sm btn-outline-danger"
onclick="deleteProduct(${product.id})">Delete</button>

            </td>

        </tr>

    `).join("")}

</tbody>

</table>

</div>

` : '<p class="text-center">No products found. Add some products to get
started.</p>'}

</div>

</div>

`;

}

// Load user dashboard

```

```

function loadUserDashboard(container) {

  const products = getProducts();

  const cart = getCart();

  container.innerHTML = `

    <div class="d-flex justify-content-between align-items-center mb-4">

      <h2>User Dashboard</h2>

      <span class="badge bg-primary">${cart.reduce((total, item) => total +
item.quantity, 0)} Items in Cart</span>

    </div>

    <div class="row">

      <div class="col-md-4 mb-4">

        <div class="card dashboard-card text-center">

          <div class="card-body">

            <div class="feature-icon text-primary">

              <i class="fas fa-shopping-bag"></i>

            </div>

            <h5 class="card-title">Browse Products</h5>

            <p class="card-text">Explore our product catalog</p>

            <button class="btn btn-primary" onclick="loadPage('product-list')">View
Products</button>

          </div>

        </div>

      </div>

      <div class="col-md-4 mb-4">

        <div class="card dashboard-card text-center">

          <div class="card-body">

```



```

        <div class="feature-icon text-warning">
            <i class="fas fa-shopping-cart"></i>
        </div>

        <h5 class="card-title">Your Cart</h5>

        <p class="card-text">Manage your shopping cart</p>

        <button class="btn btn-warning" onclick="loadPage('cart')">View
Cart</button>

    </div>

</div>

</div>

```

```

<div class="col-md-4 mb-4">

    <div class="card dashboard-card text-center">

        <div class="card-body">

            <div class="feature-icon text-success">

                <i class="fas fa-cash-register"></i>

            </div>

            <h5 class="card-title">Checkout</h5>

            <p class="card-text">Complete your purchase</p>

            <button class="btn btn-success" onclick="loadPage('checkout')"
${cart.length === 0 ? 'disabled' : ''}>Checkout</button>

        </div>

    </div>

</div>

</div>

```

```

<div class="card dashboard-card mt-4">

    <div class="card-header bg-primary text-white">

        <h5 class="mb-0">Featured Products</h5>

```

```

</div>

<div class="card-body">

  <div class="row">

    ${products.slice(0, 3).map(product => `

      <div class="col-md-4 mb-3">

        <div class="card product-card h-100">

          

          <div class="card-body">

            <h5 class="card-title">${product.name}</h5>

            <p class="card-text">${product.description}</p>

            <p class="fw-bold">${product.price.toFixed(2)}</p>

            <button class="btn btn-primary btn-sm"
onclick="addToCart(${product.id})">Add to Cart</button>

          </div>

        </div>

      `).join("")}

    </div>

  </div>

</div>

</div>

```

```
// Load add product page
```

```

function loadAddProductPage(container) {

  container.innerHTML = `

    <div class="row justify-content-center">

      <div class="col-md-8">

```

```
<div class="card dashboard-card">

  <div class="card-header bg-primary text-white">

    <h3 class="card-title mb-0">Add New Product</h3>

  </div>

  <div class="card-body">

    <form id="add-product-form">

      <div class="mb-3">

        <label for="product-name" class="form-label">Product Name</label>

        <input type="text" class="form-control" id="product-name" required>

      </div>

      <div class="mb-3">

        <label for="product-description" class="form-
label">Description</label>

        <textarea class="form-control" id="product-description" rows="3"
required></textarea>

      </div>

      <div class="mb-3">

        <label for="product-price" class="form-label">Price</label>

        <input type="number" class="form-control" id="product-price"
step="0.01" min="0" required>

      </div>

      <div class="mb-3">

        <label for="product-image" class="form-label">Image URL</label>

        <input type="url" class="form-control" id="product-image" required>

      </div>

      <button type="submit" class="btn btn-primary">Add Product</button>

      <button type="button" class="btn btn-secondary"
onclick="loadPage('admin-dashboard')">Cancel</button>

    </form>

  </div>

</div>
```

```
        </div>
    </div>
</div>
</div>
`;
```

```
document.getElementById('add-product-form').addEventListener('submit',
function(e) {
    e.preventDefault();

    const name = document.getElementById('product-name').value;
    const description = document.getElementById('product-description').value;
    const price = parseFloat(document.getElementById('product-price').value);
    const image = document.getElementById('product-image').value;

    addProduct({ name, description, price, image });

    alert('Product added successfully!');
    loadPage('admin-dashboard');
});
}
```

```
// Load product list page
function loadProductListPage(container) {
    const user = getCurrentUser();
    const products = getProducts();

    container.innerHTML = `
```

```

<div class="d-flex justify-content-between align-items-center mb-4">
  <h2>${user.role === 'admin' ? 'Manage Products' : 'Product Catalog'}</h2>
  ${user.role === 'admin' ?
    `<button class="btn btn-primary" onclick="loadPage('add-product')">
      <i class="fas fa-plus me-1"></i>Add Product
    </button>` :
    `<span class="badge bg-primary">${products.length} Products</span>`}
  }
</div>

```

```

<div class="row">
  ${products.map(product => `
    <div class="col-md-4 mb-4">
      <div class="card product-card h-100">
        
        <div class="card-body">
          <h5 class="card-title">${product.name}</h5>
          <p class="card-text">${product.description}</p>
          <p class="fw-bold">${product.price.toFixed(2)}</p>

          ${user.role === 'admin' ? `
            <div class="btn-group w-100">
              <button class="btn btn-outline-primary btn-sm"
onlick="editProduct(${product.id})">Edit</button>

              <button class="btn btn-outline-danger btn-sm"
onlick="deleteProduct(${product.id})">Delete</button>
            </div>
          ` : ``}
        </div>
      </div>
    `
  )}

```

```
<button class="btn btn-primary btn-sm"
onclick="addToCart(${product.id})">Add to Cart</button>
```

}

</div>

</div>

</div>

```
`).join("")}
```

</div>

 \therefore

}

```
// Load cart page
```

```
function loadCartPage(container) {
```

```
const cart = getCart();
```

```
const total = cart.reduce((sum, item) => sum + (item.price * item.quantity), 0);
```

```
container.innerHTML = `
```

```
<div class="d-flex justify-content-between align-items-center mb-4">
```

Shopping Cart

`${cart.reduce((total, item) => total + item.quantity, 0)} Items`

</div>

```
{cart.length > 0 ? `
```

```
<div class="table-responsive">
```

Category	Item	Value
Electronics	Smartphone	1200
Electronics	Laptop	1500
Electronics	Tablet	800
Electronics	Smartwatch	300
Electronics	Headphones	150
Electronics	Smart TV	1000
Electronics	Gaming Console	400
Electronics	Smart Home Hub	100
Electronics	Wireless Earbuds	120
Electronics	Smart Lock	200
Electronics	Smart Doorbell	150
Electronics	Smart Light Bulbs	50
Electronics	Smart Thermostat	180
Electronics	Smart Plug	30
Electronics	Smart Camera	100
Electronics	Smart Scale	40
Electronics	Smart Mirror	250
Electronics	Smart Refrigerator	1200
Electronics	Smart Washing Machine	800
Electronics	Smart Dryer	700
Electronics	Smart Dishwasher	600
Electronics	Smart Oven	500
Electronics	Smart Microwave	400
Electronics	Smart Coffee Maker	150
Electronics	Smart Toaster	80
Electronics	Smart Kettle	60
Electronics	Smart Blender	100
Electronics	Smart Juicer	120
Electronics	Smart Air Purifier	200
Electronics	Smart Dehumidifier	150
Electronics	Smart Humidifier	100
Electronics	Smart Fan	80
Electronics	Smart Heater	120
Electronics	Smart Radiator	150
Electronics	Smart Radiator Controller	50
Electronics	Smart Radiator Valve	30
Electronics	Smart Radiator Thermostat	100
Electronics	Smart Radiator Sensor	20
Electronics	Smart Radiator Switch	10
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics	Smart Radiator Sensor	5
Electronics	Smart Radiator Switch	5
Electronics	Smart Radiator Plug	5
Electronics	Smart Radiator Outlet	5
Electronics	Smart Radiator Inlet	5
Electronics	Smart Radiator Drain	5
Electronics	Smart Radiator Filter	5
Electronics	Smart Radiator Valve	5
Electronics		

|
 Product |

```

        <th>Price</th>

        <th>Quantity</th>

        <th>Total</th>

        <th>Action</th>

    </tr>

</thead>

<tbody>

    ${cart.map(item => `

        <tr>

            <td>

                

                ${item.name}

            </td>

            <td>${item.price.toFixed(2)}</td>

            <td>${item.quantity}</td>

            <td>${(item.price * item.quantity).toFixed(2)}</td>

            <td>

                <button class="btn btn-sm btn-outline-danger"

onclick="removeFromCart(${item.productId})">

                    <i class="fas fa-trash"></i>

                </button>

            </td>

        </tr>

    `).join("")}

</tbody>

<tfoot>

    <tr>

        <td colspan="3" class="text-end fw-bold">Total:</td>

```

```

        <td colspan="2" class="fw-bold">${total.toFixed(2)}</td>

    </tr>

</tfoot>

</table>

</div>

<div class="d-flex justify-content-between mt-4">

    <button class="btn btn-outline-secondary" onclick="loadPage('product-
list')">Continue Shopping</button>

    <button class="btn btn-success" onclick="loadPage('checkout')">Proceed to
Checkout</button>

</div>

`:`

<div class="text-center py-5">

    <i class="fas fa-shopping-cart fa-3x text-muted mb-3"></i>

    <h3>Your cart is empty</h3>

    <p>Browse our products and add something to your cart</p>

    <button class="btn btn-primary" onclick="loadPage('product-list')">Browse
Products</button>

</div>

    }

    ;

}

```

```
// Load checkout page
```

```
function loadCheckoutPage(container) {

    const cart = getCart();

    const total = cart.reduce((sum, item) => sum + (item.price * item.quantity), 0);

```


container.innerHTML = `

```
<div class="row">
```

```
<div class="col-md-8">
```

```
<div class="card dashboard-card mb-4">
```

```
<div class="card-header bg-primary text-white">
```

```
<h3 class="card-title mb-0">Checkout</h3>
```

```
</div>
```

```
<div class="card-body">
```

```
<form id="checkout-form">
```

```
<h5 class="mb-3">Shipping Information</h5>
```

```
<div class="row">
```

```
<div class="col-md-6 mb-3">
```

```
<label for="first-name" class="form-label">First Name</label>
```

```
<input type="text" class="form-control" id="first-name" required>
```

```
</div>
```

```
<div class="col-md-6 mb-3">
```

```
<label for="last-name" class="form-label">Last Name</label>
```

```
<input type="text" class="form-control" id="last-name" required>
```

```
</div>
```

```
</div>
```

```
<div class="mb-3">
```

```
<label for="address" class="form-label">Address</label>
```

```
<input type="text" class="form-control" id="address" required>
```

```
</div>
```

```
<div class="row">
```

```
<div class="col-md-6 mb-3">
```

```
<label for="city" class="form-label">City</label>
```

```
<input type="text" class="form-control" id="city" required>
```

</div>

<div class="col-md-6 mb-3">

<label for="zip" class="form-label">ZIP Code</label>

<input type="text" class="form-control" id="zip" required>

</div>

</div>

<hr class="my-4">

<h5 class="mb-3">Payment Information</h5>

<div class="mb-3">

<label for="card-name" class="form-label">Name on Card</label>

<input type="text" class="form-control" id="card-name" required>

</div>

<div class="mb-3">

<label for="card-number" class="form-label">Card Number</label>

<input type="text" class="form-control" id="card-number"
placeholder="XXXX XXXX XXXX XXXX" required>

</div>

<div class="row">

<div class="col-md-6 mb-3">

<label for="expiration" class="form-label">Expiration Date</label>

<input type="text" class="form-control" id="expiration"
placeholder="MM/YY" required>

</div>

<div class="col-md-6 mb-3">

<label for="cvv" class="form-label">CVV</label>

<input type="text" class="form-control" id="cvv" placeholder="XXX"
required>

</div>

</div>

<hr class="my-4">

<button type="submit" class="btn btn-success w-100">Complete
Purchase</button>

</form>

</div>

</div>

</div>

<div class="col-md-4">

<div class="card dashboard-card">

<div class="card-header bg-primary text-white">

<h5 class="mb-0">Order Summary</h5>

</div>

<div class="card-body">

`\${cart.map(item =>`

<div class="d-flex justify-content-between mb-2">

<div>\${item.name} x\${item.quantity}</div>

<div>\${(item.price * item.quantity).toFixed(2)}</div>

</div>

`).join(")}</div>

<hr>

<div class="d-flex justify-content-between fw-bold">

```

        <div>Total:</div>

        <div>$$${total.toFixed(2)}</div>

    </div>

</div>

</div>

</div>

</div>
`
;

```

```

document.getElementById('checkout-form').addEventListener('submit', function(e) {

    e.preventDefault();

    // In a real application, you would process the payment here

    alert('Order placed successfully! Thank you for your purchase.');
```

clearCart();

loadPage('user-dashboard');

```

});

}

```

```
// ===== HELPER FUNCTIONS =====
```

```
// Logout function
```

```

function logout() {

    setCurrentUser(null);

    loadPage('welcome');

}

```

```
// Edit product function
```

```
function editProduct(productId) {
```

```

const products = getProducts();

const product = products.find(p => p.id === productId);

if (!product) return;

const container = document.getElementById('main-content');
container.innerHTML = `
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card dashboard-card">
        <div class="card-header bg-primary text-white">
          <h3 class="card-title mb-0">Edit Product</h3>
        </div>
        <div class="card-body">
          <form id="edit-product-form">
            <input type="hidden" id="edit-product-id" value="${product.id}">
            <div class="mb-3">
              <label for="edit-product-name" class="form-label">Product
Name</label>
              <input type="text" class="form-control" id="edit-product-name"
value="${product.name}" required>
            </div>
            <div class="mb-3">
              <label for="edit-product-description" class="form-
label">Description</label>
              <textarea class="form-control" id="edit-product-description"
rows="3" required>${product.description}</textarea>
            </div>
            <div class="mb-3">

```

```

        <label for="edit-product-price" class="form-label">Price</label>

        <input type="number" class="form-control" id="edit-product-price"
value="${product.price}" step="0.01" min="0" required>

    </div>

    <div class="mb-3">

        <label for="edit-product-image" class="form-label">Image
URL</label>

        <input type="url" class="form-control" id="edit-product-image"
value="${product.image}" required>

    </div>

    <button type="submit" class="btn btn-primary">Update
Product</button>

    <button type="button" class="btn btn-secondary"
onclick="loadPage('product-list')">Cancel</button>

    </form>

</div>

</div>

</div>

</div>

`;

```

```

    document.getElementById('edit-product-form').addEventListener('submit',
function(e) {

    e.preventDefault();

    const id = parseInt(document.getElementById('edit-product-id').value);
    const name = document.getElementById('edit-product-name').value;
    const description = document.getElementById('edit-product-description').value;
    const price = parseFloat(document.getElementById('edit-product-price').value);
    const image = document.getElementById('edit-product-image').value;

```

```
    updateProduct({ id, name, description, price, image });

    alert('Product updated successfully!');

    loadPage('product-list');

  });
}
```

// Delete product function

```
function deleteProduct(productId) {
  if (confirm('Are you sure you want to delete this product?')) {
    deleteProduct(productId);
    alert('Product deleted successfully!');
    loadPage('product-list');
  }
}
```

// Add to cart function

```
function addToCart(productId) {
  const products = getProducts();
  const product = products.find(p => p.id === productId);

  if (product) {
    addToCart(product);
  }
}
```

// Remove from cart function

```
function removeFromCart(productId) {  
    if (confirm('Are you sure you want to remove this item from your cart?')) {  
        removeFromCart(productId);  
    }  
}
```

```
// Initialize the application
```

```
function initApp() {  
    loadNavigation();  
    loadPage('welcome');  
}
```

```
// Start the application when the page loads
```

```
window.onload = initApp;
```

```
</script>
```

```
</body>
```

```
</html>
```