Ecommerce application

Fejlesztői dokumentáció – frontend

Komponensek

cart-details

A vásárlói kosár tartalmának megjelenítésére szolgál egy e-kereskedelmi webalkalmazásban. A kód Angular direktívákat és bindingokat használ az adatok dinamikus kezelésére és megjelenítésére, beleértve a termékadatokat, mennyiségeket és árakat.

Főbb Komponensek és Funkciók

HTML Szerkezet

- Táblázat (table): A termékek listáját egy táblázat formájában jeleníti meg. A táblázat három oszloppal rendelkezik: termék képe, termék részletei, és műveletek (pl. mennyiség módosítása, eltávolítás).

Dinamikus Tartalomkezelés (Angular)

- ngIf direktíva: Az *ngIf="cartItems.length > 0" ellenőrzi, hogy vannak-e termékek a kosárban. Ha nincsenek, egy figyelmeztető üzenet jelenik meg.
- ngFor direktíva: Az *ngFor="let tempCartItem of cartItems" egy ciklus, amely minden kosárban lévő termékhez külön sor generál a táblázatban.

Termék Adatok

- Kép: Minden termékhez tartozik egy kép (<img src="{{ tempCartItem.imageUrl }}">), amely a termék vizuális reprezentációja.
- Név és Ár: A termék neve és egységára, az ár pénznem formátumra konvertálva (currency: 'USD').

Műveletek

- Mennyiség Kezelése: A felhasználó növelheti vagy csökkentheti a termék mennyiségét a + és - gombokkal. A mennyiség módosításai a incrementQuantity(tempCartItem) és decrementQuantity(tempCartItem) függvényekhez kötöttek.
- Termék Eltávolítása: A remove(tempCartItem) gombbal lehet eltávolítani a terméket a kosárból.
- Alösszeg Számítása: Minden termék alösszege kiszámításra kerül (tempCartItem.quantity * tempCartItem.unitPrice), és megjelenik az adott sorban.

Végösszeg és Egyéb Információk

- Teljes Mennyiség és Ár: Az összes termék teljes mennyisége és a végösszeg az utolsó sorban jelenik meg, valamint egy gomb, amely a pénztár oldalra navigál (routerLink="/checkout").

Stílusok és Elrendezés

- CSS Osztályok: A Bootstrap keretrendszer osztályait (table, btn, btn-primary, stb.) használja a vizuális megjelenés és elrendezés formázásához.

Példa Használat

A felhasználó hozzáad termékeket a kosárhoz, amelyek megjelennek a táblázatban. A termékek mennyiségét a felhasználó a + és - gombok segítségével állíthatja be, és szükség esetén eltávolíthat egy terméket a Remove gombbal. Amikor a kosár üres, egy figyelmeztető üzenet jelenik meg.

A CartDetailsComponent egy Angular komponens, amely a src/app/components mappában található. Ez a komponens felelős a vásárlók kosárának részleteinek megjelenítéséért, beleértve a termékek listáját, azok mennyiségét, és az összesített árat.

Komponens Deklaráció

- Selector: A komponens selector-a app-cart-details, amelyet az Angular template-ekben használhatunk a komponens beillesztéséhez.
- TemplateUrl: A cart-details.component.html fájl tartalmazza a komponens HTML struktúráját.
- StyleUrls: A cart-details.component.css fájl definiálja a komponenshez kapcsolódó CSS stílusokat.

Főbb Tulajdonságok

- cartItems (CartItem[]): Egy tömb, amely a kosárban lévő termékeket tárolja.
- totalPrice (number): A kosár teljes ára.
- totalQuantity (number): A kosárban lévő termékek összmennyisége.

Konstruktor és Függőségek

- CartService: A CartService szolgáltatást a konstruktor injektálja. Ez a szolgáltatás kezeli a kosár adatokat és biztosítja a komponens számára a szükséges funkciókat, mint például a termékek hozzáadása, eltávolítása, és a mennyiségük módosítása.

Életciklus Metódusok

- ngOnInit(): Az Angular inicializálási fázisában hívódik meg. Itt hívódik meg a listCartDetails() metódus, amely frissíti a komponens állapotát a kosárral kapcsolatos adatokkal.

Metódusok

listCartDetails

- Leírás: Frissíti a cartItems, totalPrice, és totalQuantity tulajdonságokat az aktuális értékekkel a CartService segítségével.
- Folyamat:
  Lekéri a kosár termékeit a CartService-ből.
  Feliratkozik a totalPrice és totalQuantity observable-ökre, hogy értesítést kapjon azok változásairól.
  Meghívja a computeCartTotals() metódust a CartService-ben, amely kiszámítja a kosár teljes árát és mennyiségét.

incrementQuantity

- Paraméterek:
  theCartItem (CartItem): Az a termék, amelynek a mennyiségét növelni kell.
- Leírás: Növeli a megadott termék mennyiségét a kosárban.

decrementQuantity

- Paraméterek:
- theCartItem (CartItem): Az a termék, amelynek a mennyiségét csökkenteni kell.
- Leírás: Csökkenti a megadott termék mennyiségét a kosárban.

remove

- Paraméterek:
- theCartItem (CartItem): Az a termék, amelyet eltávolítanak a kosárból.
- Leírás: Eltávolítja a megadott terméket a kosárból.

cart-status

A CartStatusComponent egy Angular komponens, amely a src/app/components mappában helyezkedik el. Ez a komponens felelős az e-kereskedelmi weboldal kosár állapotának megjelenítéséért, beleértve a teljes vásárlási összeget és a termékek számát a kosárban.

Komponens Deklaráció

- Selector: app-cart-status - Ezt a selectort használjuk az Angular template-ekben a komponens beillesztéséhez.
- TemplateUrl: cart-status.component.html - Itt található a komponens HTML struktúrája.
- StyleUrls: cart-status.component.css - Itt definiálták a komponenshez kapcsolódó CSS stílusokat.

Főbb Tulajdonságok

- totalPrice (number): A kosárban lévő összes termék teljes ára. Kezdeti értéke 0.00 USD.
- totalQuantity (number): A kosárban lévő összes termék száma. Kezdeti értéke 0.

Konstruktor és Függőségek

- CartService: Ez a szolgáltatás kezeli a kosárral kapcsolatos adatokat, mint a termékek összesített ára és mennyisége. A CartService példányát a konstruktor injektálja.

Életciklus Metódusok

- ngOnInit(): Az Angular inicializálási fázisában hívódik meg. Ebben a metódusban hívódik meg az updateCartStatus() függvény, amely frissíti a komponens állapotát a kosár aktuális adatokkal.

Metódusok

updateCartStatus

- Leírás: Ez a metódus frissíti a totalPrice és totalQuantity értékeket a CartService által biztosított aktuális adatok alapján.
- Folyamat:
  - TotalPrice Feliratkozás: Feliratkozik a CartService totalPrice observable-jára, és frissíti a totalPrice értéket a kosár aktuális teljes árával.
  - TotalQuantity Feliratkozás: Hasonlóképpen, feliratkozik a totalQuantity observable-ra, és frissíti a totalQuantity értéket a kosárban lévő termékek számával.

checkout

A CheckoutComponent egy kulcsfontosságú Angular komponens az e-kereskedelmi alkalmazásban, amely a vásárlási és fizetési folyamatot kezeli. Ez a komponens lehetővé teszi a felhasználók számára, hogy megadjanak szállítási és számlázási címeket, valamint hitelkártya-adatokat, és véglegesítsék a vásárlást.

Komponens Deklaráció

- Selector: app-checkout
- TemplateUrl: checkout.component.html - A komponens HTML sablonja.
- StyleUrls: checkout.component.css - A komponens stíluslapjai.

Főbb Attribútumok

- checkoutFormGroup (FormGroup): Űrlap csoport az adatok gyűjtéséhez és validálásához.
- totalPrice (number): A kosár teljes ára.
- totalQuantity (number): A kosárban lévő termékek teljes mennyisége.
- creditCardYears (number[]) és creditCardMonths (number[]): Elérhető hitelkártya évek és hónapok.
- countries (Country[]): A rendelkezésre álló országok listája.
- shippingAddressStates (State[]) és billingAddressStates (State[]): Az országokhoz tartozó államok vagy tartományok listái.

Konstruktor és Függőségek

- FormBuilder: Formok létrehozására és validálására használt Angular szolgáltatás.
- Luv2ShopFormService: Országok, államok és hitelkártya adatok lekérdezésére használt szolgáltatás.
- CartService: A kosáradatok kezelésére használt szolgáltatás.
- CheckoutService: A vásárlási folyamat API-n keresztüli kezelésére használt szolgáltatás.
- Router: Navigáció kezelésére használt Angular szolgáltatás.

Életciklus Metódusok

- ngOnInit(): Inicializálja a komponenst, létrehozza a szükséges form csoportokat, és betölti az alapadatokat (pl. országok, hitelkártya információk).

Metódusok és Funkciók

- reviewCartDetails(): Frissíti a totalPrice és totalQuantity értékeket a kosárszolgáltatásból.
- onSubmit(): Beküldi a vásárlási adatokat, ellenőrzi a form validitását, összeállítja a Purchase objektumot, és meghívja a CheckoutService-t a vásárlás véglegesítéséhez.
- resetCart(): Törli a kosár tartalmát és visszaállítja a formot a sikeres vásárlás után.
- handleMonthsAndYears(): Kezeli a hitelkártya lejárati hónapjainak dinamikus frissítését az évek változásával.
- getStates(formGroupName: string): Lekéri az adott országhoz tartozó államokat a form megfelelő részéhez.
- copyShippingAddressToBillingAddress(event): Másolja a szállítási címadatokat a számlázási címbe, ha a felhasználó ezt választja.

Service-ek

cart-service

A CartService egy Angular szolgáltatás, amely az e-kereskedelmi alkalmazás kosár funkcióit kezeli, beleértve a termékek hozzáadását és eltávolítását a kosárból, valamint a kosár teljes árának és mennyiségének számítását.

Szolgáltatás Deklaráció

- ProvidedIn: root - Ez azt jelenti, hogy a szolgáltatás globálisan elérhető az alkalmazásban.

Főbb Tulajdonságok

- cartItems (CartItem[]): A kosárban lévő termékek listája.
- totalPrice (Subject<number>): A kosár teljes árát tartalmazó reaktív adatforrás.
- totalQuantity (Subject<number>): A kosárban lévő termékek teljes mennyiségét tartalmazó reaktív adatforrás.

Metódusok és Funkciók

addToCart

- Paraméterek:
  theCartItem (CartItem): A kosárhoz hozzáadandó termék.
- Leírás: Ellenőrzi, hogy a termék már szerepel-e a kosárban. Ha igen, növeli a mennyiséget. Ha nem, hozzáadja a terméket a kosárhoz.
- Működés: A termék azonosítója alapján keressük meg a terméket a kosárban. Ha megtaláljuk, növeljük a mennyiséget; egyébként új elemként adjuk hozzá.

computeCartTotals

- Leírás: Kiszámítja a kosár teljes árát és a teljes mennyiséget.
- Működés: Végigmegy a kosár összes elemén, összeadja a termékek által generált részösszegeket és mennyiségeket, majd frissíti a totalPrice és totalQuantity adatforrásokat az új értékekkel.

decrementQuantity

- Paraméterek:
  theCartItem (CartItem): A mennyiségét csökkenteni kívánt termék.
- Leírás: Csökkenti a megadott termék mennyiségét a kosárban. Ha a mennyiség nulla lesz, eltávolítja a terméket.

remove

- Paraméterek:
  theCartItem (CartItem): A kosárból eltávolítandó termék.
- Leírás: Eltávolítja a megadott terméket a kosárból.
- Működés: Megkeresi a termék indexét a kosárban, és eltávolítja azt. Ezután újraszámolja a kosár teljes árát és mennyiségét.

logCartData

- Paraméterek:
- totalPriceValue (number): A kosár jelenlegi teljes ára.
- totalQuantityValue (number): A kosár jelenlegi teljes mennyisége.
- Leírás: Naplózza a kosár jelenlegi állapotát a konzolon debugolási célokból. Kiírja minden termék nevét, mennyiségét, egységárát és részösszegét, valamint a teljes árat és mennyiséget.

checkout-service

A CheckoutService egy Angular szolgáltatás az e-kereskedelmi alkalmazásban, amely a vásárlási tranzakciók feldolgozásáért felel. A szolgáltatás a backend szerverrel kommunikál az API-n keresztül, hogy elküldje a vásárlás adatait és kezelje a tranzakció válaszát.

Szolgáltatás Deklaráció

- ProvidedIn: root - Globális szinten biztosítja a szolgáltatást az alkalmazásban, ami optimalizálja a memória használatot és lehetővé teszi a szolgáltatás egyszerű injektálását bárhol az alkalmazásban.

Főbb Tulajdonságok

- purchaseUrl (string): Az API végpont URL-je, amely a vásárlási tranzakciókat kezeli. Jelen esetben ez /api/checkout/purchase.

Konstruktor és Függőségek

- HttpClient: Az Angular HttpClient modulja, amely lehetővé teszi a HTTP kérések küldését a webes API-khoz. A HttpClient segítségével a CheckoutService elküldi a vásárlási adatokat a szervernek.

Metódusok és Funkciók

placeOrder

- Paraméterek:
  purchase (Purchase): A vásárlás adatait tartalmazó objektum, amely a szükséges információkat küldi a szervernek a tranzakció feldolgozásához.
- Visszatérési Típus: Observable<any> - Az aszinkron művelet eredményét egy Observable-ben adja vissza, amely lehetővé teszi a felhasználói felület számára, hogy reagáljon a tranzakció sikerességére vagy hibájára.

- Működés: Meghívja az HttpClient post metódusát, amely az adott purchaseUrl címen POST kérést indít a purchase objektummal. A választ egy Observable formájában kapja meg, amit a komponensek feliratkozhatnak kezelni.

product-service

A ProductService egy Angular szolgáltatás, amely az e-kereskedelmi alkalmazás termékkezelési funkcióit kezeli. Ez a szolgáltatás felelős a termékek és termékkategóriák lekérdezéséért az API-n keresztül, beleértve az oldalazást is, ami a nagy terméklisták kezelését teszi hatékonyabbá.

Szolgáltatás Deklaráció

- ProvidedIn: root - Globálisan elérhetővé teszi a szolgáltatást az alkalmazásban.

Főbb Tulajdonságok

- baseUrl (string): Az API alap URL-je a termékekhez.
- categoryUrl (string): Az API alap URL-je a termékkategóriákhoz.

Konstruktor és Függőségek

- HttpClient: Az Angular HttpClient modulja, amely lehetővé teszi HTTP kérések küldését a webes API-khoz.

Metódusok és Funkciók

getProduct

- Paraméterek:
  - theProductId (number): A lekérdezendő termék azonosítója.
- Visszatérési Típus: Observable<Product>
- Leírás: Lekér egy terméket azonosító alapján.
- Működés: A HttpClient get metódusát használja a specifikus termék URL-jével.

getProductListPaginate

- Paraméterek:
  - thePage (number): Az oldalszám.
  - thePageSize (number): Egy oldal mérete.
  - theCategoryId (number): A termékkategória azonosítója.
- Visszatérési Típus: Observable<GetResponseProducts>
- Leírás: Lekér egy oldalnyi terméket a megadott kategóriából, oldalazással.
- Működés: Lekérdezési URL összeállítása és adatok lekérése.

getProductList

- Paraméterek:
  - theCategoryId (number)
- Visszatérési Típus: Observable<Product[]>
- Leírás: Lekér minden terméket egy adott kategóriából.
- Működés: Hasonló a paginált verzióhoz, de oldalazás nélkül.

searchProducts

- Paraméterek:
- theKeyword (string)
- Visszatérési Típus: Observable<Product[]>

- Leírás: Termékek keresése kulcsszó alapján.
- Működés: A keresési kulcsszóval összeállított URL-ről kér le adatokat.

searchProductsPaginate

- Paraméterek:
- thePage (number)
- thePageSize (number)
- theKeyword (string)
- Visszatérési Típus: Observable<GetResponseProducts>
- Leírás: Keresett termékek oldalazott listájának lekérése.
- Működés: Oldalazott lekérdezés kulcsszó alapján.

getProductCategories

- Visszatérési Típus: Observable<ProductCategory[]>
- Leírás: Lekér minden elérhető termékkategóriát.
- Működés: A categoryUrl használata a kategóriák lekérésére, a válasz mappelése a megfelelő struktúrára.