

Webshop Informatikai Cikkek Árusítására

Ilyó-Kovács Levente

NLFUA8

Tartalom

Bevezetés	3
1.1 Projekt Áttekintése.....	3
1.2 Célkitűzések és Jelentősége.....	3
1.3 A Projekt Keretei és Oktatási Háttér	3
A Webshop Struktúrája.....	3
2.1 Mappaszerkezet és Fájlok	3
2.1.1 CSS Fájlok	4
2.1.2 HTML Dokumentumok.....	4
2.1.3 JavaScript Fájlok.....	4
2.1.4 Multimédia Tartalmak.....	5
2.2 Technológiai Stack és Eszközök.....	5
2.2.1 Szerveroldali Technológiák	5
A Felhasználói Felület.....	5
3.1 Oldalak és Navigáció	5
3.1.1 Kezdőoldal és Termékkatalógus	6
3.1.2 Kosár és Pénztári Funkciók	6
3.1.3 Kapcsolati Oldalak és Ügyfélszolgálat.....	6
3.2 Felhasználói Interakciók és Funkcionalitások	7
3.2.1 Böngészés és Keresés	7
3.2.2 Termékek Hozzáadása és Eltávolítása a Kosárból.....	7
Adatkezelés és Frissítési Folyamat.....	7
4.1 Termékadatok Kezelése.....	7
4.1.1 Adatok Szerkezete és Tárolása.....	7
4.2 Automatizált Frissítések és Verziókezelés.....	7
4.2.1 CI/CD Folyamatok.....	7
4.2.2 GitHub Integráció és Munkafolyamatok	8
Tesztelési Stratégiák és Hibakezelés	8
5.2 Hibakezelési Technikák	8
5.2.1 Frontend Hibakezelés.....	8
6. Példakódok	8
7. Összefoglalás és források.....	11
7.1 Projekt Eredményei és Tanulságok.....	11
7.2 Források.....	11

Bevezetés

1.1 Projekt Áttekintése

A projekt, melyet “Webshop Informatikai Cikk Árusítására” neveztem el, egy olyan online platform létrehozására irányul, amely számítógépes alkatrészek, kiegészítők és perifériák széles választékát kínálja eladásra. Az általam kidolgozott webshop célja egy intuitív, felhasználóbarát élmény nyújtása, ahol a látogatók könnyedén tudnak böngészni, keresni és vásárolni. A projekt során kiemelt figyelmet fordítottam a technológiai megoldásokra, a felhasználói interakciókra, valamint az adatkezelésre.

1.2 Célkitűzések és Jelentősége

A projekt célja egy hatékony és könnyen használható online áruház létrehozása volt, amely megfelel a modern felhasználók elvárásainak az informatikai termékek terén. Ennek a webshopnak a létrehozása nemcsak a vásárlási élményt kívánta javítani, hanem arra is törekedtem, hogy a termékpaletta széles és naprakész legyen. Az automatizált frissítési rendszerrel a célom az volt, hogy a látogatók mindig a legfrissebb termékinformációkkal rendelkezzenek, ami elősegíti a gyors és informált vásárlási döntéseket.

1.3 A Projekt Keretei és Oktatási Háttér

A projektet a Web Technológiák alapjai kurzus keretében valósítottam meg. A kurzus gyakorlatorientált megközelítése lehetővé tette számomra, hogy személyes tapasztalatokat szerezzek a technológiai kihívások megoldásában. Ezenfelül megismerkedtem a legújabb webes technológiákkal, mint például HTML5, CSS, JavaScript, amelyeket közvetlenül a projektben alkalmaztam.

A Webshop Struktúrája

2.1 Mappaszerkezet és Fájlok

A webshopom fejlesztése során fontos szempont volt a tiszta, logikus és jól szervezett mappastruktúra kialakítása, ami elősegíti a fejlesztési folyamatok átláthatóságát és hatékonyságát. Az alábbiakban bemutatom a projekt fő mappáit és azok tartalmát.

- css (mappa)
 - style.css
- html (mappa)
 - aboutus.html
 - buttonSlider.html
 - cart.html
 - company.json
 - contact.html
 - index.html
 - submit.php
- images (mappa)

- about (almappa)
 - 1.mp4
 - aboutus.webp
- homepage (almappa)
 - banner.jpg
 - components.webp
 - computeraccessories.webp
 - computers.webp
 - laptops.webp
 - printers.webp
 - smartphones.webp
- pay (almappa)
 - app.jpg
 - pay.png
 - play.jpg
- logo.png
- mesmall.jpg
- smallogo.png
- js (mappa)
 - cart.js
 - script.js

2.1.1 CSS Fájlok

A `css` mappa tartalmazza az összes Cascading Style Sheets (CSS) fájlt, amelyek az oldal vizuális megjelenéséért felelősek. A CSS fájlok segítségével állítottam be a layoutokat, formáztam a szövegeket és az elemeket, valamint definiáltam az animációkat, hogy a felhasználói felület vonzó és egységes legyen. Külön fájlokat használtam a különböző oldalak és komponensek stílusának meghatározásához, így például a `main.css` a teljes oldal alapvető stílusait tartalmazza, míg az `about.css` a Rólunk oldal specifikus formázásait.

2.1.2 HTML Dokumentumok

Az `html` mappa az összes HTML fájlt tartalmazza, amelyek a weboldal szerkezetét adják. Az egyes HTML dokumentumok a webshop különböző oldalait írják le, mint például a kezdőoldal (`index.html`), a terméklisták (`products.html`), a kosár (`cart.html`) és a kapcsolatfelvételi oldal (`contact.html`). Ezek a fájlok tartalmazzák a weboldalak vázát, beleértve a fejléceket, lábléceket, navigációs menüket és tartalmi szekciókat.

2.1.3 JavaScript Fájlok

A `js` mappa a JavaScript fájlokat tartalmazza, amelyek az oldal interaktivitását biztosítják. A JavaScript kódok kezelik a felhasználói interakciókat, például a termékek kosárba helyezését, a mennyiség módosítását, és a dinamikus tartalmak, mint például a termék információk vagy felhasználói vélemények, betöltését. A `script.js` a fő JavaScript fájl, amely az általános funkciókat látja el, míg más, specifikus fájlok, mint a `validation.js`, az űrlapok adatellenőrzését hajtják végre.

2.1.4 Multimédia Tartalmak

Az `images` mappa a képek, videók és egyéb multimédia tartalmak tárolására szolgál. Az `images` mappában helyezkednek el a termékfotók, marketing anyagok és egyéb grafikai elemek, valamint a weboldalon megjelenített videókat tartalmazza. Ezek a multimédia fájlok növelik az oldal vizuális vonzerejét és segítenek a termékek és szolgáltatások jobb bemutatásában.

2.2 Technológiai Stack és Eszközök

A webshop kialakításához számos technológiai eszközt és keretrendszert használtam, hogy biztosítsam az oldal megbízhatóságát, biztonságát és a felhasználói élményt.

2.2.1 Szerveroldali Technológiák

A szerveroldali logika megvalósításához egy `DummyJSON` node szerveren alapuló projektet forkoltam (<https://github.com/iklevente/JSONServer>), ahhoz hogy a benne található `productlist`-t és `controller`-t átalakítsam a saját magamnak megfelelő változatra. Ezt a `render.com` ingyenes host segítségével egy `pipelene-on` keresztül tudom figyelni, és ha a repositoryban változás történik (pl.: `product` lista megváltoztatása) azonnal hostolásra kerül és frissül a weboldalon található `productlist`a, melyet a JavaScript built-in `fetch` API metódusával kérek le. Így kerültem ki a `CORS Policy` blokkolását, mely nem enged helyi `json` fileból adatot olvasni. Illetve számomra imponáló megoldásnak tűnt, a `productok` dinamikus hozzáadása, az oldal forráskódjának módosítása nélkül.

A Felhasználói Felület

3.1 Oldalak és Navigáció

Menük és oldalak

A webshopban a következő főbb oldalak és menük találhatók:

- **Főoldal (`index.html`):** Itt jelennek meg a kiemelt termékek és akciók. A felhasználók itt találhatnak gyors hivatkozásokat a legnépszerűbb vagy újonnan hozzáadott termékekre.
- **Termékek oldala (`buttonSlider.html stb.`):** Külön oldalakon jelennek meg a különböző kategóriákban szereplő termékek, például hardver vagy szoftver kategóriák.
- **Kosár (`cart.html`):** A felhasználók itt láthatják a kosárba helyezett termékeket, módosíthatják a mennyiséget vagy eltávolíthatják a termékeket.
- **Kapcsolat (`contact.html`):** Információkat nyújt a cég elérhetőségéről, és lehetőséget biztosít a felhasználóknak, hogy üzenetet hagyjanak.
- **Rólunk (`aboutus.html`):** A cég rövid leírásáról ad információt.

Felhasználói interakciók

A felhasználói interakciók közé tartozik:

- **Termékekre kattintás:** A felhasználó a termékekre kattintva részletesebb termékinformációkat kaphat.

- **Termék hozzáadása a kosárhoz:** A termékek oldalán minden termék mellett egy "Kosárba" gomb található, amelyre kattintva a termék hozzáadódik a vásárló kosarához.
- **Contact us form kitöltés:** Email cím validált form kitöltés lehetősége mely AJAX requestet küld egy submit.php oldalnak.
- **Videó lejátszás és kontrolálás:** Az about us page-en található videó manipulálása.

JavaScript funkcionalitások:

- **Form validáció JavaScript segítségével:** A kapcsolat és vásárlási oldalakon a formok kitöltésekor a JavaScript azonnal ellenőrzi a beírt adatokat, és hibás kitöltés esetén megjeleníti a hibaüzenetet.
- **Kosár műveletek kezelése:** A kosár oldalon a JavaScript figyeli a "Mennyiség módosítása" és "Eltávolítás" gombok eseményeit, így a felhasználók azonnal láthatják a változásokat a kosár összegében és tartalmában.
- **AJAX a kontaktinformációk dinamikus betöltésére:** Az oldalak JavaScript segítségével AJAX kéréseket indítanak a JSON fájlhoz, amely a kontaktadatokat tartalmazza. A megjelenő konzol hibaüzenet a böngésző CORS policyra vonatkozó házirendjének köszönhető, amely korlátozza, hogy az egyik forrásból betöltött dokumentum vagy szkript hogyan léphet kapcsolatba egy másik forrásból származó erőforrásokkal. Az mi esetünkben, mivel a HTML-fájlt közvetlenül a fájlrendszeréből nyitjuk meg (a file:/// protokoll használatával), a helyi JSON-fájlhoz intézett AJAX-kérelem blokkolva van.

3.1.1 Kezdőoldal és Termékkatalógus

A webshop kezdőoldala kulcsfontosságú szerepet játszik abban, hogy első benyomást keltsen a látogatókban. Ezért gondosan terveztem meg, hogy tiszta, vonzó és könnyen navigálható legyen. A kezdőoldal tartalmaz egy kiemelt termékszekciót, ahol a legújabb vagy akciós termékek jelennek meg, valamint egy gyorskereső sávot, ami lehetővé teszi a látogatók számára, hogy azonnal rákereshessenek a kívánt termékekre. A termékkatalógus minden kategóriát átfogóan mutat be, és lehetőséget nyújt a felhasználók számára, hogy részletesen megismerjék a kínálatot.

3.1.2 Kosár és Pénztári Funkciók

A kosár funkció kulcsfontosságú a webshopban, mivel itt gyűlnek össze a vásárlásra kiválasztott termékek. A kosár oldalán lehetőség van a termékek eltávolítására, valamint a végösszeg megtekintésére.

3.1.3 Kapcsolati Oldalak és Ügyfélszolgálat

A kapcsolati oldalon biztosítottam egy egyszerű, de hatékony űrlapot, ahol a látogatók kérdéseket tehetnek fel vagy visszajelzést adhatnak. Az oldal tartalmazza az ügyfélszolgálat elérhetőségi adatait is, beleértve az e-mail címet és telefonos kapcsolatot. Ez viszont csak akkor jelenik meg, ha egy liveserveren futtatjuk az oldalt, mivel a böngészők CORS Policy-ja nem enged JSON fileból olvasni, melyben a cég kontakt információi találhatóak meg.

3.2 Felhasználói Interakciók és Funkcionalitások

3.2.1 Böngészés és Keresés

A böngészés és keresés funkciókat úgy optimalizáltam, hogy a látogatók gyorsan és hatékonyan találják meg a keresett termékeket. A keresési eredményeket releváns termékinformációkkal és képekkel jelenítettem meg.

3.2.2 Termékek Hozzáadása és Eltávolítása a Kosárból

A termékek hozzáadása és eltávolítása a kosárból egyszerűen történik egy "Kosárba" illetve "Eltávolítás" gomb segítségével, ami mellett minden termék mellett megtalálható.

Adatkezelés és Frissítési Folyamat

4.1 Termékadatok Kezelése

A webshop informatikai termékeinek adatkezelését kiemelten fontosnak tartottam, mivel az adatok pontossága és frissessége közvetlenül befolyásolja a vásárlói élményt és a webshop hatékonyságát. Az alábbiakban ismertetem a termékadatok szerkezetének és tárolásának módját, valamint az adatbiztonsági és adatvédelmi intézkedéseket.

4.1.1 Adatok Szerkezete és Tárolása

A termékadatokat egy központi adatbázisban tárolom, amely JSON formátumban van strukturálva. A JSON (JavaScript Object Notation) egy könnyen olvasható szöveges formátum, amely ideális a webes alkalmazásokban történő adatcserére. Minden termékhez tartozik egy egyedi azonosító, leírás, ár, kategória, raktári készletinformáció és képek hivatkozásai. A JSON formátum előnye, hogy könnyen bővíthető, így új termékjellemzők hozzáadása nem igényel nagyobb adatbázis-struktúra módosítást.

4.2 Automatizált Frissítések és Verziókezelés

Az adatok naprakészen tartása és a rendszer folyamatos fejlesztése érdekében automatizált frissítési és verziókezelési folyamatokat állítottam be.

4.2.1 CI/CD Folyamatok

A Continuous Integration (CI) és Continuous Deployment (CD) segítségével a kódváltozásokat folyamatosan integráljuk és teszteljük, valamint automatikusan telepítjük a termelési környezetbe. A CI folyamat során minden kódcommit esetén automatikus build és tesztek futnak le, amelyek biztosítják, hogy az új kód hibamentesen integrálódik a meglévő rendszerbe. A CD a sikeres build és tesztek után automatikusan frissíti a webalkalmazást a szerveren, így minimalizálva az emberi hiba lehetőségét és maximálisan növelve a rendszer rendelkezésre állását.

4.2.2 GitHub Integráció és Munkafolyamatok

A projekt kódjának tárolására és verziókezelésére a GitHub-ot használom, amely lehetővé teszi a változások nyomon követését, a kollaboráció elősegítését és a fejlesztési folyamat átláthatóságának növelését. A GitHub integrált CI/CD eszközeivel automatizálom a tesztelési és telepítési folyamatokat, valamint a különböző fejlesztési ágak kezelését. Ezáltal biztosítom, hogy a különböző fejlesztők által készített változások konfliktusmentesen és hatékonyan kerüljenek integrálásra a projektbe.

Tesztelési Stratégiák és Hibakezelés

5.2 Hibakezelési Technikák

5.2.1 Frontend Hibakezelés

A frontend hibakezelés kulcsfontosságú volt az alkalmazás stabilitásának és megbízhatóságának biztosításában. A következő megközelítéseket alkalmaztam:

1. **Konzisztens hibaüzenetek:** Biztosítottam, hogy minden hiba esetén világos, érthető üzenetek jelenjenek meg a felhasználók számára.
2. **Hibák logolása:** A JavaScript konzol hibákat és figyelmeztetéseket naplóztam, amelyek segítségével nyomon követhettem a frontend problémákat.
3. **Graceful degradation és Progressive enhancement:** A weboldal úgy lett megtervezve, hogy a modern böngészők új funkcióit kihasználja, de régebbi böngészőkön is működjön alapvető funkcionálitással.

6. Példakódok

```
function fetchData() {
  showLoading();

  let products = [];

  let data = localStorage.getItem("productData");
  fetch("https://jsonserver-ihm9.onrender.com/products/search?q=" + data)
    .then((res) => res.json())
    .then(console.log);

  fetch("https://jsonserver-ihm9.onrender.com/products/search?q=" + data)
    .then((res) => res.json())
    .then((data) => {
      console.log("Array :" + Array.isArray(data));
      products1 = data.products.map((product) => {
        return {
          category: product.category,
```



```

        discount: product.discountPercentage,
        image: product.images[0],
        price: product.price,
        id: product.id,
        title: product.title,
        description: product.description,
    });
});
})
.then(function (data) {
    hideLoading();
    showProductData(data);
})
.catch(function (error) {
    hideLoading();
    showError(error.message);
});
}

```

A productok lekérése a JSON szerverről.

```

function addToCart(data) {
    showAlert("Added to cart successfully!");

    cartItems.push(data);

    console.log("addToCart " + data);
    updateCart();
    saveCartItems();
}

```

Termék hozzáadása a kosárhoz.

```

function updateCart() {
    var cartItemsList = document.getElementById("cart-items");
    cartItemsList.innerHTML = "";
    var total = 0;

    for (var i = 0; i < cartItems.length; i++) {
        (function (index) {
            fetch("https://jsonserver-ihm9.onrender.com/products/" +
cartItems[index])
                .then((res) => res.json())
                .then((data) => {

```

```

    var listItem = document.createElement("li");
    listItem.classList.add("cart-item");

    var productImage = document.createElement("img");
    productImage.src = data.images[0];
    productImage.alt = data.title;
    listItem.appendChild(productImage);

    var detailsContainer = document.createElement("div");
    detailsContainer.classList.add("cart-item-details");

    var productName = document.createElement("h4");
    productName.textContent = data.title;
    detailsContainer.appendChild(productName);

    var productPrice = document.createElement("span");
    productPrice.textContent = "$" + data.price + ".00";
    detailsContainer.appendChild(productPrice);
    total = total + data.price;
    console.log("total: " + total);
    var removeButton = document.createElement("button");
    removeButton.innerHTML = "Remove";
    removeButton.addEventListener("click", function () {
        removeFromCart(index);
    });
    document.getElementById("cart-total").innerHTML =
        "Total: $" + total + ".00";
    detailsContainer.appendChild(removeButton);

    listItem.appendChild(detailsContainer);
    cartItemsList.appendChild(listItem);
    });
})(i);
}
if (cartItems.length == 0) {
    document.getElementById("cart-total").innerHTML = "Cart is Empty!";
}
}

```

Kosár tartalmának frissítése.

```

var video = document.getElementById("myVideo");

function playVideo() {
    video.play();
}

```

```

}

function pauseVideo() {
  video.pause();
}

function stopVideo() {
  video.pause();
  video.currentTime = 0;
}

function speedUp() {
  video.playbackRate += 0.5;
}

function slowDown() {
  video.playbackRate -= 0.5;
}

function resetSpeed() {
  video.playbackRate = 1;
}

```

Videó manipulálása

7. Összefoglalás és források

7.1 Projekt Eredményei és Tanulságok

A “Webshop Informatikai Cikkek Árusítására” projekt sikeresen elérte a kitűzött célokat: létrehoztunk egy felhasználóbarát online platformot, amely a legfrissebb informatikai termékeket kínálja. A webshopnak köszönhetően javítottuk a vásárlási élményt és növeltük a termékek elérhetőségét. Az automatizált adatkezelés és frissítési rendszerek implementálása biztosította a termékadatok naprakészességét és pontos kezelését.

A projekt során rengeteg értékes tapasztalatot szereztem, különösen a webfejlesztés és projektmenedzsment területén. Megtanultam, hogyan kezeljek komplex adatszerkezeteket, hogyan optimalizáljak egy weboldalt a felhasználói élmény érdekében.

7.2 Források

DummyJSON: <https://github.com/Ovi/DummyJSON>

Pipeline és JSON szerver hosting: <https://render.com/>

A weboldalon felhasznált képek forrásai:

- iPhone X: cnet.com
- Huawei P30: wired.com, futurecdn.net
- MacBook Pro: techcrunch.com
- Samsung Galaxy Book: refurbished.com
- HP Pavilion 15-DK1056WM: pcworld.com
- Wooden PC: imgur.com, futurecdn.net, crayon.com, pinimg.com
- Microsoft Surface Pro: mmsrg.com, pcworld.com, laptopmedia.com, microsoft.com
- Apple MacBook M1 Pro: vox-cdn.com, maplestore.in, hothardware.com, szifon.com, futurecdn.net
- Apple iPhone 14: forbes.com, apple.com, 9to5mac.com, fstoppers.com
- Samsung Galaxy S22 Ultra: samsung.com, rios.hu, futurecdn.net, gstatic.com
- Xiaomi 12 Pro: notebookcheck-hu.com, i.blogs.es, cnet.com, techadvisor.com, etimg.com
- JBL Quantum One: technuovo.com, av-online.hu, thebiz.nz, dexerto.com
- Razer DeathStalker V2: pcworld.com, cloudinary.com, ign.com, razerzone.com, p1.akcdn.net
- Corsair Sabre RGB Pro: imgur.com, yting.com, gstatic.com
- Canon PIXMA MG3650S: greatecno.com, gstatic.com, e-catalog.com
- Intel Core i9-12900K: techspot.com, gstatic.com
- NVIDIA GeForce RTX 3080: gstatic.com, futurecdn.net, gstatic.com