

Szegedi SZC Tóth János Mórahalmi Szakképző Iskola és Szilágyi Mihály  
Kollégium

## **VIZSGAREMEK**

Vásár helyfoglalási és eladó keresési rendszer fejlesztése

### **Készítette:**

Kiss-Torma Krisztina  
Paska-Fáth Mariann  
Tanács Kata

Szoftverfejlesztő és -tesztelő képzés

Mórahalom

2024

## Tartalomjegyzék

1. Bevezetés	4
2. Fejlesztői dokumentáció	6
2.1. Fejlesztő környezet	6
2.1.1. Hardver	6
2.1.2. Szoftver	7
2.1.3. Fejlesztői Eszközök	7
2.2. Adatszerkezet	8
2.2.1. Userdata tábla	8
2.2.2. Date_market tábla	9
2.2.3. Place tábla	10
2.2.4. Product tábla	10
2.2.5. Reservation kapcsoló tábla	11
2.2.6. Product range kapcsoló tábla	12
2.3. Az oldal felépítése és főbb algoritmusai	13
2.3.1. Bejelentkezés	13
2.3.2. Kijelentkezés	13
2.3.3. Regisztráció	14
2.3.4. Profil	15
2.3.5. Oldalsó menü	17
2.3.6. Adminisztrációs felületek	18
2.3.7. Árusok listája	21
2.3.8. Keresés	21
2.3.9. Helyfoglalási kérelem	22
2.3.10. Helyfoglalásaim	25
2.4. Tesztdokumentáció	28
2.5. Fejlesztési lehetőségek	29
3. Felhasználói dokumentáció	30
3.1. Szükséges hardver eszközök és szoftverek	30
3.2. A program részletes bemutatása	30
3.2.1. Bejelentkezés	30
3.2.2. Regisztráció	31
3.2.3. Profil	33
3.2.4. Oldalsó menü	34
3.2.5. Keresés	34
3.2.6. Árusok listája	36
3.2.7. Helyfoglalási kérelem	37
3.2.8. Helyfoglalásaim	38
3.2.9. Adminisztrátor - Felhasználó kezelés	39
3.2.10. Adminisztrátor - Helyfoglalás kezelése	40

3.2.11. Adminisztrátor - Új termék kategória	41
3.2.12. Adminisztrátor - A következő vásár beállítása	42
3.3. Helytelen használatból adódó hibajelzések magyarázata	43
3.3.1. Üres mezők hibái:	43
3.3.2. Formátum ellenőrzések:	43
3.3.3. Karakter hossz ellenőrzések	43
3.3.4. Termék kategória ellenőrzés	44
3.3.5. Általános hibák	44
3.4. Információkérés lehetőségeinek megadása	44
4. Összefoglalás	44
5. Köszönetnyilvánítás	45
6. Irodalomjegyzék	45

## 1. Bevezetés

Projektünk egy olyan összetett weboldal megalkotását tűzte ki célul, amely dinamikus és interaktív platformként szolgál a vásárlók és árusok számára egyaránt. Az alapgondolat egy karácsonyi ajándékvadászat során fogant meg, amikor is egy korábban több alkalommal, különböző helyszíneken megpillantott kézműves terméket szerettünk volna megvásárolni, de nem volt információnk arról, hogy az árus az adott napon jelen van-e, illetve hol található. Így született meg a Vásár weblap gondolata, melynek segítségével rá tudunk keresni egy adott termék kategóriára és meg tudjuk nézni, hogy az mely vásárokon és mely elárúsító helyeken található. Úgy véltük, hogy egy ilyen kereső funkció pótolhatná a hiányosságokat a modern, sietős világ és a tradicionális piaci forgatag összefonódásában. Ezt az elképzelést tovább szélesítettük egy online helyfoglalási lehetőséggel az árusok részére.

A gondolat megszületése után piackutatást végeztünk, hogy kiderítsük, léteznek-e hasonló kezdeményezések. Kutatásaink során arra a következtetésre jutottunk, hogy a vásári témában nem jöttek létre hasonló weboldalak a környezetünkben. A témánkhoz legközelebb álló weboldalak piaci portálok voltak, mint például a szegedi Bödön Piac vagy az Újpesti Piac weboldalai, de ezek is eltérő kérdéskörök köré csoportosultak.

Weboldalunk lehetőséget biztosít a felhasználóknak, hogy felfedezzék a kívánt termékeket és azok eladóit, miközben a vásári térkép segítségével könnyedén azonosíthatják azok pontos helyét. Ezen kívül a regisztrált árusok a következő vásárokra foglalhatnak helyet, mely kérelmeket az adminisztrátorok értékelik ki. A moderátor felelős a újonnan regisztrált felhasználók fiókjainak aktiválásáért, továbbá joga van a felhasználók fiókjainak felfüggesztésére vagy törlésére, amennyiben azok nem megfelelően használják az oldal nyújtotta lehetőségeket vagy megsértik a vásár házirendjét. Ő jogosult új termékkategóriák felvételére is, valamint a következő vásári időpontok meghatározására. A weboldal keresési funkciója regisztráció nélkül is hozzáférhető, így az információk széles körben elérhetőek.

A weboldal tervezésekor kiemelt figyelmet fordítottunk a felhasználóbarát interfész kialakítására. A kezdőlapra egy beágyazott Google térkép segíti a távolabbról érkező látogatókat a vásár pontos helyének megtalálásában, ezzel is csökkentve a navigációs

nehézségeket. A felület designja modern és letisztult, ami hozzájárul a felhasználói élmény növeléséhez.

A projekt célcsoportjai a vásár látogatói és az árusok. Ezt szem előtt tartva fejlesztettük ki a platformot. A felhasználói felület Bootstrap alapú, ami a reszponzív dizájn biztosításához nélkülözhetetlen.

Összefoglalva, a projekt egy olyan digitális megoldást kínál, amely híd szerepet tölt be a hagyományos vásárok és a modern online kereskedelmi platformok között.

## 2. Fejlesztői dokumentáció

### 2.1. Fejlesztő környezet

A projekt kialakításához szükséges fejlesztői környezet magában foglalja a hardveres és szoftveres infrastruktúrát, valamint azokat az eszközöket, amelyek elengedhetetlenek a kódolási és tesztelési folyamatokhoz.

#### 2.1.1. Hardver

A fejlesztéshez használt hardverek a standard teljesítményű számítógépek, amelyek képesek támogatni a fejlesztési folyamatokat anélkül, hogy teljesítménybeli korlátokba ütköznének. A projekt során 3 számítógép és egy mobiltelefont segítségével zajlott a tesztelés. Ezek hardver konfigurációit az alábbiakban ismertetjük:

##### 1. Számítógép:

- Felhasználó: Kiss-Torma Krisztina
- Operációs rendszer: Microsoft Windows 11
- Böngésző: Brave v1.64.122-Chromium 123.0.6312.122 (64 bites)
- Kijelző felbontása: 1920x1080

##### 2. Számítógép:

- Felhasználó: Paska-Fáth Mariann
- Operációs rendszer: Microsoft Windows 11 Home
- Böngésző: Google Chrome v24.0.6367.62 (64 bites)
- Kijelző felbontása: 1920x1080

##### 3. Számítógép:

- Felhasználó: Tanács Kata
- Operációs rendszer: Microsoft Windows 10 Pro (x64)
- Böngésző: Google Chrome v121.0.6167.161(64 bites)
- Kijelző felbontása: 1920x1080

##### 4. Mobiltelefon:

- Felhasználó: Kiss-Torma Krisztina
- Típus: iPhone12

- Operációs rendszer: iOS 17.4.1
- Böngésző: Safari
- Kijelző felbontása: 2532 x 1170

### 2.1.2. Szoftver

A számítógépeken Windows operációs rendszerek futnak, amely egy megbízható és széles körben elterjedt platform a fejlesztők körében. Az alkalmazás szerverként XAMPP-ot (v3.3.0) használunk, amely egy népszerű és könnyen kezelhető Apache disztribúció, tartalmazva a MySQL adatbázist (MariaDB v10.4.32) és a PHP (v8.2.12) futtató környezetet. Ez a szerverkörnyezet biztosítja, hogy a helyi fejlesztési és tesztelési folyamatok zökkenőmentesen, az élő környezethez hasonló feltételek mellett zajlanak.

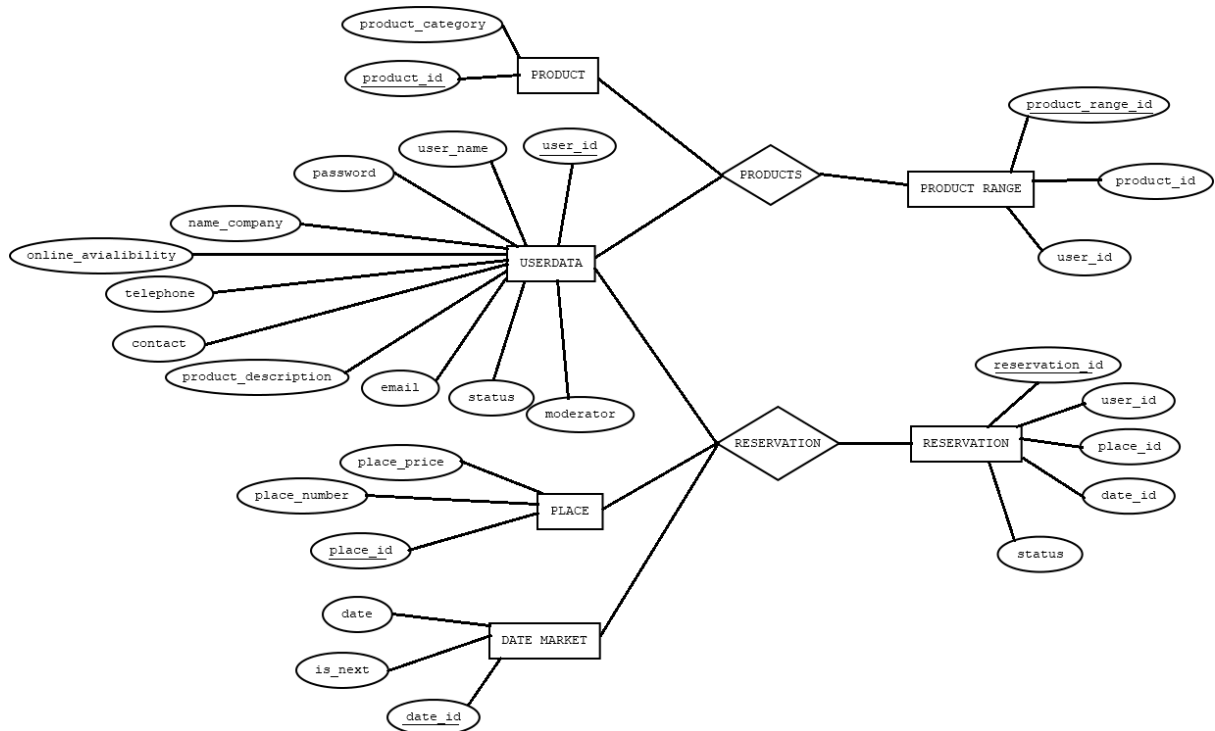
### 2.1.3. Fejlesztői Eszközök

A forráskód szerkesztéséhez a Visual Studio Code-ot használjuk, amely egy könnyen használható, de rendkívül hatékony integrált fejlesztői környezet (IDE), amely támogatja a PHP, HTML, CSS, JavaScript és számos egyéb programozási nyelv szintaxisát és fejlesztési folyamatait. A Visual Studio Code számos bővítményt kínál, amelyek lehetővé teszik a fejlesztők számára, hogy testre szabják a környezetet saját igényeiknek megfelelően, növelve ezzel a produktivitást és az együttműködést. A projekt verziókövetése a Git rendszeren keresztül történik, amely egy elterjedt és hatékony eszköz a kódváltozások nyomon követésére és kezelésére. A Git lehetővé teszi a fejlesztők számára, hogy biztonságosan dolgozzanak közösen a kódokon, elősegítve a hatékony együttműködést és a visszaállíthatóságot.

Ezek a technológiák és eszközök biztosítják, hogy a fejlesztői csapat hatékonyan és zökkenőmentesen dolgozhasson a projekt különböző elemein, garantálva ezzel a weboldal magas szintű teljesítményét és stabilitását.

## 2.2. Adatszerkezet

Az adatbázis logikai modelljét egyed-kapcsolat jelölésrendszer segítségével vizualizáltuk, mely az 1. ábrán látható.



1. ábra. Adatbázis modell egyed-kapcsolat diagram

A munka során olyan adatbázist építettünk fel, mely alkalmas információt tárolni piacunkról, az oldalon regisztrált felhasználókról, az oldalon tett helyfoglalásaikról. A megvalósításhoz 4 adattáblára, 2 kapcsolatra, és 2 kapcsolótáblára volt szükség. A továbbiakban az imént felsorolt elemek kerülnek bemutatásra.

### 2.2.1. Userdata tábla

Ez a tábla a piac adatbázisának egyik fontos része, melyben a felhasználók személyes adatai tárolódnak. Ez a tábla lehetővé teszi a felhasználók azonosítását, a profiladatok kezelését. A táblába kerülés során automatikusan generálódik egy azonosító. A felhasználó név mező unique, ami biztosítja hogy az adatbázisba nem kerülhet két azonos felhasználó név. A jelszavakat hashelve tároljuk az adatbázisban, ami azt jelenti, hogy az eredeti jelszavakat speciális kriptográfiai eljárással átalakítjuk, és az átalakított értékeket mentjük el. Ezáltal az adatbázisban nem tároljuk a



felhasználók tiszta szövegű jelszavait, ami biztonsági előnyt jelent, mivel azok nem olvashatóak közvetlenül az adatbázisban tárolt formájukban.

A tábla tartalmazza még a felhasználó fontosabb, egy vásár szempontjából releváns adatait, mint cégnév, kontakt személy, e-mail cím, telefonszám, online elérhetőség, valamint egy leírást az általa árusított termékekről, tevékenységi köréről. A felhasználó regisztráció során kiválasztja az általa árusított termék kategóriákat is, amellyel a keresést szeretnénk pontosabbá, és felhasználó baráttá tenni. Tartalmaz még egy status mezőt, amely információt ad a felhasználó jelenlegi státuszáról: újonnan jelentkezett, aktív, függőben lévő vagy törölt-e a felhasználó. A moderator mező azt tárolja, hogy a felhasználó moderátor vagy sem.

NÉV	TÍPUS	JELLEMZŐ
user_id	int(11)	PRIMARY KEY, AUTO INCREMENT
user_name	varchar(20)	NOT NULL, UNIQUE
password	varchar(80)	NOT NULL
name_company	varchar(50)	NOT NULL
contact	varchar(50)	NOT NULL
telephone	varchar(20)	NOT NULL
email	varchar(50)	NOT NULL
online_availability	varchar(500)	DEFAULT NULL
product_description	varchar(300)	NOT NULL
moderator	tinyint(1)	NOT NULL DEFAULT 0
status	tinyint(1)	NOT NULL DEFAULT 0

#### 2.2.2. Date\_market tábla

A *date\_market* tábla az egyes vásárok időpontjait tárolja, melyek dátum formátumban vannak rögzítve. Minden új rekord hozzáadásakor automatikusan generálódik egy egyedi azonosító a táblában. Emellett a tábla tartalmaz egy *is\_next*

nevű mezőt, amelynek értéke lehet 0 vagy 1. Az *is\_next* mező arra utal, hogy az adott dátum az aktuális időpontban következő vásárhoz kapcsolódik-e vagy sem. Amennyiben az érték 1, az azt jelenti, hogy az adott dátum a következő vásár időpontja, míg ha az érték 0, akkor az adott dátum nem a következő vásárhoz tartozik.

NÉV	TÍPUS	JELLEMZŐ
date_id	int(11)	PRIMARY KEY, AUTO INCREMENT
date	date	NOT NULL
is_next	tinyint(1)	NOT NULL DEFAULT 0

### 2.2.3. Place tábla

A *place* tábla az adott vásár összes helyét tárolja. Minden új hely rekordjának létrehozásakor automatikusan generálódik egy egyedi azonosító a táblában. A tábla tartalmazza a helyek sorszámát, amelyet a felhasználó a helyfoglalási kérelmek menüpontban található térképről tud leolvasni. Emellett a tábla tartalmazza a helyek árát is, így lehetővé teszi az alkalmazás számára, hogy kezelje a helyfoglalási díjakat és információkat. A *place* tábla struktúrája segít az alkalmazásnak a vásárokhöz kapcsolódó helyek hatékony és strukturált tárolásában, valamint azokhoz való könnyű hozzáférésben és kezelésben.

NÉV	TÍPUS	JELLEMZŐ
place_id	int(11)	PRIMARY KEY, AUTO INCREMENT
place_number	varchar(10)	NOT NULL
place_price	int(11)	NOT NULL

### 2.2.4. Product tábla

A *product* tábla a termékkategóriák azonosítására és kezelésére szolgál, lehetővé téve az alkalmazás számára, hogy kezelje és megjelenítse azokat a felhasználók

számára. Minden új hely rekordjának létrehozásakor automatikusan generálódik egy egyedi azonosító a táblában.

NÉV	TÍPUS	JELLEMZŐ
product_id	int(11)	PRIMARY KEY, AUTO INCREMENT
product_category	varchar(30)	NOT NULL, UNIQUE

#### 2.2.5. Reservation kapcsoló tábla

A *reservation* tábla egy kapcsolótábla, amely a foglalások nyilvántartására szolgál az almazásban. A tábla struktúrája lehetővé teszi az egyes foglalásokhoz kapcsolódó információk hatékony és strukturált tárolását. Segítségével az alkalmazás széleskörű funkcionalitást biztosít a foglalások kezelésére és nyomon követésére.

Minden foglalás egyedi azonosítót kap, amely automatikusan generálódik, és ez a tábla elsődleges kulcsa.

A *user\_id* oszlop tárolja a foglalást végző felhasználó azonosítóját. Ez egy külső kulcs (foreign key), amely kapcsolódik a felhasználók táblájához, így azonosítja, hogy mely felhasználóhoz tartozik a foglalás.

A *place\_id* oszlop tárolja a foglalt hely azonosítóját. Szintén külső kulcs, amely kapcsolódik a helyek táblájához, így azonosítja, hogy mely helyet foglalták le.

A *date\_id* oszlop tárolja a foglalás dátumának azonosítóját. Külső kulcsként kapcsolódik a dátumok táblájához, így meghatározza, hogy melyik dátumhoz tartozik a foglalás.

A *status* oszlop jelzi a foglalás aktív vagy inaktív állapotát. Alapértelmezett értéke 0, ami inaktív, jóváhagyásra váró állapotot jelent. Az admin hatásköre a foglalást jóváhagyni, ekkor a status 1-es értéket kap.

NÉV	TÍPUS	JELLEMZŐ
reservation_id	int(11)	PRIMARY KEY, AUTO INCREMENT

user_id	int(11)	NOT NULL, FOREIGN KEY
place_id	int(11)	NOT NULL, FOREIGN KEY
date_id	int(11)	NOT NULL, FOREIGN KEY
status	tinyint(1)	DEFAULT 0

#### 2.2.6. Product range kapcsoló tábla

A *product\_range* tábla egy kapcsolótábla, amely összekapcsolja a felhasználókat (user\_data tábla) és a termékeket (product tábla) a vásár alkalmazásban. Ennek a táblának a struktúrája lehetővé teszi a termékkínálatok hatékony kezelését és tárolását az adott felhasználókhoz rendelve. Mivel egy felhasználó több terméket is kínálhat a vásáron, és egy termék több felhasználóhoz is kapcsolódhat, ez a sok-sok kapcsolat strukturált és hatékony kezelését igényel az alkalmazásban.

A tábla rendelkezik egy egyedi azonosító mezővel, automatikusan növekszik, és a tábla elsődleges kulcsa.

A product\_id oszlop tárolja a termék azonosítóját, ami egy külső kulcs, és kapcsolódik a termékek (products) táblához.

A user\_id oszlop tárolja a felhasználó azonosítóját, amely egy másik külső kulcs, és kapcsolódik a felhasználók (users) táblához.

NÉV	TÍPUS	JELLEMZŐ
product_range_id	int(11)	PRIMARY KEY, AUTO INCREMENT
product_id	int(11)	NOT NULL, FOREIGN KEY
user_id	int(11)	NOT NULL, FOREIGN KEY

## 2.3. Az oldal felépítése és főbb algoritmusai

### 2.3.1. Bejelentkezés

Az oldalsó menüsorból érhető el a "Bejelentkezés" menüpont, melyre kattintva egy új oldalon lehetőség nyílik a felhasználónév és jelszó megadására. A modul a *dbconnect.php* segítségével csatlakozik az adatbázishoz, és a *session\_start()* hívással inicializálja a munkamenetet, ami elengedhetetlen a felhasználói állapotok kezeléséhez a munkamenet során.

A bejelentkezési folyamat a *submitBejelentkezes* gomb megnyomásával indul, amennyiben a felhasználói adatbáziskapcsolat (*\$dbconn*) érvényes. A felhasználónév (*felhnev*) és jelszó (*jelszo*) mezők adatait a *trim()* függvény segítségével tisztítja meg a kód, így megakadályozva felesleges szóközök bekerülését az adatok közé. Ha valamelyik adat hiányzik, a *UserLoginException* kivétel dobódik, ami jelzi a hiányzó adatokat.

A felhasználói adatok ellenőrzése az adatbázisban egy paraméterezett SQL lekérdezéssel történik, amit a *PDO::prepare* metódus hoz létre, biztosítva az SQL-injekció elleni védelmet. Amennyiben a lekérdezés nem talál pontosan egy egyezést a felhasználónévre (*user\_name*), a *UserLoginException* kivétel jelez egy hibás felhasználói azonosítót. Sikeres azonosítás és a jelszó helyességének ellenőrzése (*password\_verify()*) után, amennyiben a felhasználó státusza nem megfelelő (például felfüggesztett vagy törölt fiók), a rendszer újabb *UserLoginException* kivételeket dob megfelelő hibaüzenettel.

Sikeres bejelentkezés esetén a felhasználó adatait a munkamenetben (*\$\_SESSION*) és egy sütiben (*setcookie()*) 24 órán át tárolja a rendszer, amelyek a felhasználó azonosítását szolgálják a munkamenet során. Végül a felhasználót a kezdő oldalra irányítja.

### 2.3.2. Kijelentkezés

A kijelentkezési lehetőség az oldalsó menüsorban helyezkedik el. Amikor a felhasználó rákattint, a művelet érvényteleníti a sütiket, eltávolítja a

felhasználó-azonosítót és a felhasználó egyéb adatait a munkamenetből. Ezt követően a rendszer az index.php oldalra irányítja át.

### 2.3.3. Regisztráció

Regisztrációs folyamat:

A regisztráció során a felhasználóknak kitöltendő űrlapot kell kitölteniük, amely a bal oldali side barban található meg második menüpontként. Az űrlap az alábbi adatokat tartalmazza:

- Felhasználónév
- Email cím
- Jelszó
- Jelszó megerősítése
- Vállalkozás/cég neve
- Kapcsolattartó neve
- Telefonszám
- Online elérhetőség (opcionális)
- Termék leírása
- Termék kategória választás

Kliensoldali validáció:

A weboldal JavaScript segítségével végzi a kliensoldali validációt annak érdekében, hogy biztosítsa, hogy a felhasználó megfelelő adatokat adjon meg a regisztrációs űrlapon. A validáció magában foglalja az alábbi ellenőrzéseket:

- Kötelező mezők kitöltöttsége
- Email cím formátumának ellenőrzése
- Jelszó hosszúsága és egyezősége
- Telefonszám formátumának ellenőrzése
- Termékleírás kötelező legyen
- Termék kategória választásának ellenőrzése

#### Szerveroldali validáció:

A szerveroldali validáció a PHP segítségével történik, és további ellenőrzéseket végez annak érdekében, hogy biztosítsa az adatok helyességét és biztonságát. A szerveroldali validáció magában foglalja az alábbi ellenőrzéseket:

- Felhasználónév egyediségének ellenőrzése az adatbázisban
- Email cím egyediségének és helyes formátumának ellenőrzése
- Jelszó hosszúsága és egyezősége
- Cég nevének hosszúsága és érvényessége
- Kapcsolattartó nevének formátumának ellenőrzése
- Telefonszám formátumának ellenőrzése
- Online elérhetőség URL formátumának ellenőrzése (opcionális)
- Termékleírás kitöltésének ellenőrzése
- Termék kategória választásának ellenőrzése

#### Hibakezelés:

A regisztrációs folyamat során felmerülő hibák kezelésére különböző módszereket alkalmazunk. Ha bármely adat nem felel meg a validációs követelményeknek, a felhasználót figyelmeztetjük az adatok megfelelő kitöltésére, és megjelenítjük a hibaüzeneteket a felhasználó számára. A hibaüzenetek segítenek a felhasználónak az adatok helyesítésében és a regisztrációs folyamat sikeres befejezésében.

#### Sikeres regisztráció:

Ha a regisztrációs űrlapot sikeresen kitöltötték és az adatok megfelelnek a validációs követelményeknek, a rendszer sikeres regisztrációról tájékoztatja a felhasználót, és lehetőséget biztosít a bejelentkezésre a felhasználói fiókba.

#### 2.3.4. Profil

A felhasználói profil weboldal célja, hogy a felhasználók személyes és felhasználói adataikat megtekinthessék és módosíthassák. A weboldal lehetővé teszi a

felhasználók számára, hogy frissítsék a regisztráció során megadott adatokat, valamint hogy változtassanak jelszavukon.

Felhasználói adatok megjelenítése és módosítása:

A felhasználói adatok megjelenítése és módosítása két különálló folyamatot foglal magában, amelyek a következőket tartalmazzák:

Személyes adatok megjelenítése és módosítása:

- A felhasználói profil oldalán a felhasználók láthatják és módosíthatják a személyes adatokat, mint például az árus/cég neve, e-mail cím, kapcsolattartó neve, telefonszám és online elérhetőség.

Felhasználói adatok megjelenítése és jelszó módosítása:

- A felhasználók lehetőséget kapnak a jelenlegi jelszavuk megadására, valamint az új jelszó kiválasztására és megerősítésére a jelszó módosítása részeként.

Adatok validációja:

A felhasználói profil oldalon a kliens- és szerveroldali validáció egyaránt fontos szerepet játszik annak érdekében, hogy biztosítsuk az adatok helyességét és biztonságát.

Kliensoldali validáció:

- A weboldal JavaScript segítségével végzi a kliensoldali validációt, amely ellenőrzi az űrlapmezők kitöltöttségét és az adatok helyes formátumát.

Szerveroldali validáció:

- A PHP segítségével történik a szerveroldali validáció, amely további ellenőrzéseket végez az adatok helyességének és biztonságának biztosítása érdekében. Ide tartozik például a felhasználónév egyediségének ellenőrzése és a jelszavak biztonságos tárolása.



Hibakezelés:

A felhasználói profil oldalon felmerülő esetleges hibákat különböző módszerekkel kell kezelni annak érdekében, hogy a felhasználók zavartalanul tudják használni az oldalt.

Hibaüzenetek megjelenítése:

- Ha bármelyik validációs lépés során hiba történik, a felhasználót értesíteni kell a hibáról megfelelő hibaüzenetek segítségével.

### 2.3.5. Oldalsó menü

A bal oldalon található menü egy dinamikusan változó navigációs felület, amely a felhasználó státusza és ki/bejelentkezett állapota alapján változtatja a megjelenítendő menüpontokat.

A kód a Bootstrap *navbar-expand-lg* osztályát használja, ami lehetővé teszi, hogy a navigációs sáv dinamikusan alkalmazkodjon a különböző képernyőméretekhez. A *hamburger menü* a kisebb képernyőkön jelenik meg, lehetővé téve a felhasználók számára, hogy egy gombnyomással ki- és becsukják a navigációs menüt.

A szkriptek a `$_SESSION` globális változót használják a felhasználó bejelentkezési állapotának ellenőrzésére. Ha a felhasználó nincs bejelentkezve, a navigációs menü a "Bejelentkezés" és "Regisztráció" opciókat jeleníti meg. Bejelentkezett felhasználók esetén további menüpontok jelennek meg, mint például a "Profil", "Helyfoglalási kérelem" és "Helyfoglalásaim". Adminisztrátori jogosultságokkal rendelkező felhasználóknak speciális adminisztrációs menüpontok is elérhetőek, mint például a felhasználók kezelése, a helyfoglalások kezelése és az új termékkategóriák hozzáadása.

A navigációs sáv rendelkezik a *navbar-collapse* és *navbar-nav* osztályokkal, ezek biztosítják, hogy vertikálisan jelenjenek meg az oldalsávon belül.

### 2.3.6. Adminisztrációs felületek

#### Adminisztrátor - Felhasználó kezelés

Ez az oldal lehetővé teszi az adminisztrátorok számára a felhasználói státuszok kezelését, az adatbázisban tárolt felhasználói adatok alapján. Az alkalmazás azonosítja a bejelentkezett felhasználót mint adminisztrátort, és csak admin jogosultságokkal rendelkező felhasználók férhetnek hozzá a felülethez, ahol a felhasználói státuszokat módosíthatják.

A felületen egy űrlapon keresztül lehet kezelni a felhasználói státuszokat. A moderátornak lehetősége van az aktiválás, felfüggesztés vagy törlés műveletekre. A *setStatus* funkció kezeli a státuszok módosítását, amely egy *UPDATE* SQL utasítást hajt végre az adatbázisban a felhasználói azonosító alapján, amelyet a formról kapott adatokból vesz át.

A *generateTable* funkcióval generálható egy táblázat, amely az adott státuszú felhasználókat listázza. Ez a funkció *SELECT* lekérdezéseket használ a felhasználók adatainak lekérésére, amelyeket a felületen dinamikusan jelenít meg. Az adatok lekérdezése során a státusz paramétert a funkció paraméterként kapja meg, és az előkészített lekérdezésekben használja fel. A funkció több alkalommal is használatra kerül a felület különböző státuszú felhasználóinak listázására.

A *getUserCountForStatus* funkció segítségével lekérdezhető, hogy egy adott státuszhoz hány felhasználó tartozik, amely információ alapján a felhasználói felület gombjait aktiválhatja vagy deaktiválhatja a rendszer, attól függően, hogy van-e módosításra váró felhasználó az adott kategóriában.

A rendszer kezeli a kivételeket a felhasználói interakciók során, beleértve a *UserAdminException* és *PDOException* típusokat, amelyek információt szolgáltatnak a felhasználónak, ha hiba lép fel a műveletek során, például ha a kiválasztott felhasználó azonosítója hiányzik, vagy adatbázis kapcsolati probléma áll fenn.

## Adminisztrátor - Helyfoglalás kezelése

Ennek az adminisztrátori oldalnak a célja a helyfoglalási kérelmek kezelésének támogatása. Lehetőség van a kérelmek állapotának módosítására, beleértve azok jóváhagyására, elutasítására és aktuális állapotuk megjelenítésére. A rendszer biztonságos munkamenet-kezeléssel rendelkezik, amely csak azonosított és megfelelő jogosultságokkal rendelkező felhasználóknak engedélyezi a funkciók elérését.

Az alkalmazás a *dbconnect.php* segítségével csatlakozik az adatbázishoz, ami lehetővé teszi a további SQL műveletek végrehajtását. A kód kezeli a helyfoglalásokat az adatbázis *reservation* táblájában tárolt adatok alapján, ahol a helyfoglalások státusz szerint vannak csoportosítva. A *setReservationStatus* funkció segítségével az adminisztrátorok módosíthatják egy adott helyfoglalás státuszát.

A helyfoglalások aktuális állapotát dinamikusan jeleníti meg a rendszer a *generateTable* funkcióval, amely SQL lekérdezésekkel gyűjti össze a releváns adatokat és HTML táblázatba rendezve jeleníti meg azokat. Ez a táblázat külön nézetekben mutatja be az elbírálás alatt álló, jóváhagyott és elutasított helyfoglalásokat. Az adminisztrátorok a felhasználók nevét, a vásár dátumát és a helyszámot tekinthetik meg, ami segíti őket a helyfoglalások kezelésében.

A rendszer hibakezelést is tartalmaz, amely felismeri és kezeli az adatbázisműveletek során felmerülő kivételeket, így a *ReservationAdminException* és a *PDOException* hibákat, biztosítva ezzel az alkalmazás stabilitását és megbízhatóságát.

## Adminisztrátor - Új termék kategória

Ez az oldal biztosítja az új termékkategóriák adatbázisba való felvételét és a már létező kategóriák dinamikus megjelenítését táblázatos formában. Az alkalmazás munkamenet-alapú hitelesítést használ, így csak azonosított és megfelelő jogosultságokkal rendelkező felhasználók férhetnek hozzá.

Akinek nincs moderátori státusza azt átirányítja az index oldalra, ezzel megakadályozva a jogosulatlan hozzáférést. Az *addNewProduct* funkció előkészített SQL utasításokat használ az új termékkategória beszúrásához. Ha a termékkategória mező üres, a funkció *ProductException* kivételt dob. A *generateTable* funkció lekérdezi az adatbázisból a termékkategóriákat és egy HTML táblázatban jeleníti meg azokat.

A kód kezeli a saját kivételeket (*ProductException*) és az általános adatbázis hibákat (*PDOException*), beleértve az egyedi korlátozások megsértését és egyéb adatbázis hozzáférési hibákat. Az adminisztrátorok egy egyszerű űrlapon keresztül adhatnak meg új termékkategóriákat.

#### Adminisztrátor- A következő vásár beállítása

Ez a felugró ablak teszi lehetővé a vásári időpontok kezelését. A rendszer magában foglalja az új dátumok hozzáadását, a jelenleg aktív dátumok lekérdezését és a következő események dátumainak beállítását.

A *getNextDate* funkció a következő esemény időpontját kéri le az adatbázisból azon rekordok közül, ahol az *is\_next* jelző be van állítva. Ez az információ alapvető a felhasználói felületen történő megjelenítéshez, ahol az adminisztrátorok gyorsan megtekinthetik a következő esemény dátumát. Ha nincs következő dátum beállítva, a rendszer *null* értéket ad vissza.

Az *addNextTime* funkció lehetővé teszi új vásári dátumok hozzáadását, paraméterként megkapja a felhasználó által megadott dátumot és letárolja az adatbázisban *is\_next = false* értékkel. Ha a dátum mező üres, a *FairException* kivétel kezelés aktiválódik, figyelmeztetve a felhasználót, hogy szükséges kitölteni a mezőt.

A *generateSelect* funkció egy dinamikusan generált legördülő menüt hoz létre, amely az összes elérhető, a jövőbeni eseményekre vonatkozó dátumot tartalmazza. Ez a funkció segíti az adminisztrátorokat abban, hogy kiválasszák a következő aktív dátumot az adatbázisban már létező dátumok közül.

A *setNextDate* funkció frissíti az adatbázisban a következő eseményre vonatkozó dátumot, minden más dátumot inaktívvá téve, ezzel biztosítva, hogy csak egy

következő esemény lehessen aktív egyszerre. Ez kulcsfontosságú a rendszer integritásának megőrzéséhez, mivel elkerüli az időpontok ütközését.

Az alkalmazás továbbá kezeli a felhasználói inputokból eredő hibákat és az adatbázis-műveletek során felmerülő kivételeket is, beleértve a *PDOException* típusú hibákat, amelyek adatbázis hozzáférési problémákat jelentenek. A hibakezelés része az is, hogy a felhasználókat tájékoztatja az esetleges duplikált dátum bevitelről, amely az adatbázis egyediségi korlátozásainak megsértését jelenti.

### 2.3.7. Árusok listája

Az árusok listája egy nyilvánosan, minden érdeklődő számára elérhető tájékoztató menüpont, ahol a kiállítókat, termékeiket és elérhetőségeit lehet böngészni. A menüpontra kattintva egy táblázat jelenik meg, mely ABC rendben tartalmazza a regisztrált, és az adminisztrátor által jóváhagyott felhasználókat.

A kód PHP-ban íródott, és dinamikusan generál HTML-t a böngésző számára.

A PHP kód lekérdezést végez az adatbázisban, és a kapott eredményeket dinamikusan beilleszti az oldal tartalmába. A lekérdezés kiválasztja a kiállító nevét, termék leírását, elérhetőségét és elérhetőségi adatait a userdata táblából, ahol a státusz aktív (1). Az eredményeket egy táblázatban jeleníti meg.

A kód kezeli az esetleges hibákat, például ha nem sikerül a kapcsolódás az adatbázishoz vagy ha problémák merülnek fel a lekérdezések végrehajtása közben. A hibák kezelése segít abban, hogy az alkalmazás ne legyen sérülékeny a felhasználói élményre nézve.

### 2.3.8. Keresés

Ez a kereső funkció lehetővé teszi a felhasználók számára, hogy könnyen és hatékonyan keressenek a termékekkategóriák között a rendszerben. A pontos és gyors keresési eredmények révén a felhasználók könnyen tájékozódhatnak és részt vehetnek a kívánt eseményeken.

A keresés űrlapja a headerben található, minden látogató számára elérhető. A placeholder mutatja, hogy az oldalon termékkategóriák közti keresésre van lehetőség.

Kereső funkció: A kód a felhasználó által megadott keresőkifejezést (amely a GET metóduson keresztül érkezik az URL-ben) ellenőrzi, hogy létezik-e, és nem üres-e. Ha a kifejezés létezik és nem üres, akkor a kód elkezd a keresést az adatbázisban.

A keresési kifejezésből először levágja a felesleges szóközöket, hogy biztosítsa a tisztább adatfeldolgozást.

Ezután a kód előkészíti, majd lefuttatja az SQL lekérdezést, amely két részből áll: az aktuális és a múltbeli vásárokat vizsgálja meg a keresett termékkategória alapján.

A kód végrehajtja a lekérdezést az adatbázisban, majd megvizsgálja, hogy van-e találat. Ha találhatóak eredmények az aktuális vagy a múltbeli vásárokból a keresett termékkategóriára, akkor ezeket formázott táblázatokban jeleníti meg az oldalon. Az egyik táblázat az elmúlt vásárokat jeleníti meg, a másik a jövőbeli vásárokat mutatja. Mindkét táblázat csökkenő időrendi sorrendben mutatja a találatokat, a könnyebb áttekinthetőség érdekében.

A táblázatok azt jelenítik meg, hogy a keresett kategóriájú termékeket mely árusok árulják/árulták, mikorra és melyik helyet foglalják/foglalták.

A kód csak az adminisztrátor által jóváhagyott helyfoglalásokat jeleníti meg.

Ha nem található eredmény a keresés során, akkor a kód egy üzenetet jelenít meg arról, hogy nincs találat a keresésre. Ez az üzenet segít tájékoztatni a felhasználót, hogy nincs releváns információ az adatbázisban a keresett kifejezésre.

#### 2.3.9. Helyfoglalási kérelem

A menüpont célja, hogy a regisztrált felhasználók biztosíthassák árusító helyüket az általuk kiválasztott vásár időpontjára.

A helyfoglalás funkciója az alábbiak szerint működik:

A felhasználó választhat egy dátumot a vásárok közül egy legördülő listában, ami tartalmazza az elkövetkező vásárok időpontjait. Az időpontokat az adatbázisból lekérdezett adatok alapján töltik fel, és csak az aktuális dátumot meghaladó időpontok jelennek meg.

A következő rész HTML elemekből áll, amelyek alapvetően rejtve vannak (display: none; stílussal). Ezek közé tartozik egy címke (<h4>) és egy legördülő lista (<select>), amelyek a szabad helyeket fogják megjeleníteni az adott időpontban. Emellett van egy

"Kiválasztom" gomb (<input type="submit">), amelyet később fogunk megjeleníteni, ha a felhasználó kiválaszt egy helyet.

A JavaScript kód részlet a következőket teszi:

- Először definiálja a `vasarDatumLista` és a `szabadHelyekLista` változókat, amelyek az időpontok legördülő listáját és a szabad helyek legördülő listáját jelölik.
- A `vasarDatumLista` változóra eseményfigyelőt állít be, amely akkor fut le, amikor a felhasználó kiválaszt egy dátumot az időpontok listájából. Ebben az esetben egy AJAX kérést küld az adott dátumhoz tartozó szabad helyek lekérdezésére.
- Amikor a felhasználó kiválaszt egy dátumot a `vasarDatumLista` legördülő listájából, az eseményfigyelő meghívódik, és a kódban az XMLHttpRequest objektum segítségével küldünk egy GET kérést a szerver felé, hogy lekérjük a szabad helyek listáját a kiválasztott dátumra.
- Az XMLHttpRequest objektum működése aszinkron, ami azt jelenti, hogy a böngésző nem vár a válasza a kérés küldése után, hanem folytatja a kód futtatását. Amikor a válasz megérkezik a szerverről, az `onreadystatechange` eseménykezelő meghívódik, és a válasz állapotát ellenőrizzük. Ha a válasz sikeres (`status === 200`), akkor a kapott szabad helyeket betölti a `szabadHelyekLista` legördülő listába, majd megjeleníti mindkét elemet.
- Az aszinkron működés lehetővé teszi a felhasználói élmény javítását a weboldalon, mivel nem kell várakoznia a hosszú műveletek befejeződésére, és közben továbbra is interakcióba léphet az oldallal.
- Ha a felhasználó nem választott ki dátumot, akkor mindkét elemet elrejt.
- Az Ajax kérés a `places.php` fájl segítségével történik. A `places.php` fájl egy szerver oldali szkript, amely végrehajtja a szükséges adatbázis lekérdezéseket és válaszként visszaküldi az AJAX kérésnek a megfelelő választ. Ez a válasz tartalmazza a szabad helyeket, amelyeket a JavaScript szkript azonnal beilleszt a `szabadHelyekLista` (szabad helyek legördülő listája) elembe az oldalon, így frissítve a felhasználó által látott tartalmat a szerverről kapott adatok alapján.

- Az eseményfigyelők között van egy másik, amely figyeli a szabad helyek legördülő listában történő változásokat. Ha a felhasználó kiválaszt egy helyet, megjeleníti a "Kiválasztom" gombot.
- A "Kiválasztom" gombra kattintva az esemény figyelő átnavigálja a felhasználót a megerősítés(confirm.php) oldalra, és átadja a kiválasztott dátum és hely azonosítóját a GET paraméterekben.

Confirmation.php fájl működése:

Ez a rész a felhasználó által kiválasztott hely foglalásának véglegesítését végzi. Az oldal dinamikus, azaz a tartalom a felhasználó által történő interakciók hatására változik.

A PHP szkript a GET kérések paramétereinek kezelésével indul. Ha a megadott paraméterek megfelelőek (selectedDateld és selectedHelyld létezik), akkor lekérdezi a szükséges adatokat az adatbázisból. Ez magában foglalja a felhasználó adatait, a kiválasztott vásár dátumát és a kiválasztott hely információit. Ezeket az adatokat egy űrlapon jeleníti meg a felhasználó számára, ahol ellenőrizheti és véglegesítheti a foglalást.

A HTML form az adatokat továbbítja a save\_reservation.php fájlnak, ahol a foglalást a szerver rögzíti. Ha valamilyen hiba történik a folyamat során, akkor a hibaüzenetet kiírja az oldalra.

Save-reservation.php

A szkript először ellenőrzi, hogy a szükséges GET paraméterek (selectedDateld és selectedHelyld) léteznek-e. Ha igen, akkor a szkript lekéri a felhasználó azonosítóját a munkamenetből (session), majd az adatbázisba rögzíti a foglalást.

A foglalás rögzítése egy SQL INSERT parancs segítségével történik, amely az adott felhasználóhoz és a kiválasztott helyhez és dátumhoz társít egy foglalást. Az új foglalás alapértelmezett státusza 0, ami azt jelenti, hogy a foglalás még adminisztrátori jóváhagyásra vár.

Ha a foglalás sikeresen rögzítésre került az adatbázisban, a szkript átirányítja a felhasználót az index.php oldalra, és a foglalás sikerességét jelzi a "foglalas=1" GET paraméterrel. Ha bármely ponton hiba lép fel, az adott hibaüzenetet tárolja, és



átirányítja a felhasználót az index.php oldalra a "foglas=0" GET paraméterrel, ami a foglalás sikertelenségét jelzi.

Az index oldalon egy popup ablak ad tájékoztatást a foglalás sikerességéről/sikertelenségéről, és a további teendőkről.

Végül a exit parancs segítségével leállítja a szkript végrehajtását, hogy ne történjen további feldolgozás a kód után.

### 2.3.10. Helyfoglalásaim

A kód egy felhasználói felületet kínál az adatbázisban tárolt helyfoglalási kérelmek kezelésére és megjelenítésére, figyelembe véve azok különböző státuszait. A felhasználók így könnyen ellenőrizhetik a kérelmek állapotát és az esetleges változásokat.

A session\_start() függvényt használja a munkamenet indításához ami elengedhetetlen a felhasználói állapotok kezeléséhez a munkamenet során.

Az adatbázis lekérdezés a felhasználóhoz kapcsolódó helyfoglalási kérelmek keresését végzi az adatbázisban. A kód három különböző lekérdezést tartalmaz a kérelmek különböző státuszainak megjelenítésére: jóváhagyásra váró, jóváhagyott és törölt kérelmekre.

Minden lekérdezés az alábbi táblákból szerzi be az adatokat:

1. place: Ez a tábla tárolja a vásárok helyeit, beleértve a hely sorszámát is.
2. date\_market: A vásárok időpontjait tárolja, dátum formátumban.
3. userdata: Felhasználói adatokat tartalmaz, ideértve a felhasználó azonosítóját is.
4. reservation: A helyfoglalási kérelmeket tárolja, ideértve a kérelem státuszát is.

A lekérdezések részletesen:

- Jóváhagyásra váró kérelmek: Ez a lekérdezés kiválasztja azokat a helyfoglalási kérelmeket, amelyek státusza "0", vagyis jóváhagyásra várnak. Az eredményeket a helyek sorszámával és a vásár időpontjával együtt jeleníti meg, rendezve a vásár dátuma szerint csökkenő sorrendben.

- Jóváhagyott kérelmek: Ez a lekérdezés azokat a helyfoglalási kérelmeket választja ki, amelyek státusza "1", vagyis már jóvá lettek hagyva. Hasonlóan az előzőhöz, az eredményeket a helyek sorszámaival és a vásár időpontjával együtt jeleníti meg, rendezve a vásár dátuma szerint csökkenő sorrendben.

- Törölt kérelmek: Ez a lekérdezés azokat a helyfoglalási kérelmeket választja ki, amelyek státusza "2", vagyis már törölve lettek. Az eredményeket szintén a helyek sorszámaival és a vásár időpontjával együtt jeleníti meg, rendezve a vásár dátuma szerint csökkenő sorrendben.

Minden lekérdezés során a felhasználó azonosítóját használja feltételként ('\$user\_id' = u.user\_id), hogy csak az adott felhasználóhoz kapcsolódó kérelmeket jelenítse meg. A lekérdezések kivételkezelő blokkba vannak helyezve, hogy kezeljék az esetleges adatbázis hibákat vagy kivételeket, és megfelelő üzeneteket jelenítsenek meg a felhasználó számára.

#### 2.3.11. Stílus - A weboldal kinézetének formázása

A weboldal kinézetének kialakítása és formázása alapvető fontosságú a felhasználók megfelelő navigációjának és az általános felhasználói élmény javításának érdekében. A következő fejlesztési dokumentáció részletesen bemutatja a CSS kódokat és módszereket, amelyeket alkalmaztunk a weboldal megjelenésének formázásához.

##### Fontosabb formázási elemek

*Header:* Beállítjuk a fejléc pozícióját és szélességét, valamint elrejtjük az esetleges túlcsoorduló tartalmat. Relatív pozícióban helyezzük el az elemet.

*header img:* Segítünk beállítani a fejlécben található kép megjelenítését. A képet teljes szélességére húzzuk az oldalnak ezzel együtt arányosan növeljük/csökkentjük a magasságot. Blokk elemként jelenítjük meg.

##### Navigációs Sáv (nav-bar):

Megadjuk az alapértelmezett tulajdonságokat az unordered list (ul) elemhez. Kiürítjük a listaelemek előtti alapértelmezett pontokat és nullára állítjuk a margót és a kitöltést.

Az inline-block elemeket hozzáigazítjuk a navigációs sávhoz és blokk elemként jelenítjük meg. A navigációs sávban található hivatkozásokat paraméterezzük. Pl.: Blokk elemként való megjelenítéssel, szöveg színek beállításával, szöveg középre igazításával, text dekorációkkal.

A navigációs sávban az egér mozgatására reagáló háttér színekkel és a betűk félkövérre módosításával is foglalkozunk.

#### Oldalsáv Menü és Hover Effektus

Az oldalsáv menü a weboldalon történő navigációban elengedhetetlen és a felhasználói élményt is fokozza. A menü háttérszínét a *custom-navbar* osztállyal állítottuk be, amely a friss és barátságos #9AD768 színárnyalatot használja. Az egér fölé húzáskor kialakított hover effektus lehetővé teszi, hogy a linkek vastagabbak legyenek és az #c5f8a5 színűvé váljanak, ezzel kiemelve a kiválasztott menüpontot. Ez a vizuális jelzés segít a felhasználóknak a navigációban és javítja az oldal felhasználhatóságát.

Navigációs sáv háttérszínének is fontos szerepet tulajdonítunk.

#### Footer

A következő stílusok összességében gondoskodnak arról, hogy a lábléc megfelelően formázva legyen, a tartalom könnyen olvasható legyen, és az elrendezés érzékenyen alkalmazkodjon a különböző képernyőméretekhez.

Beállítjuk a lábléc háttérszínét, belső kitöltését, szöveg igazítását, a betűméretet, kitöltések nagyságát valamint egy specifikus részlet (langer-space) margóját a láblécben.

## 2.4. Tesztdokumentáció

A webes alkalmazás tesztelését end to end tesztekkel valósítottuk meg, melyek szimulálják a valós felhasználást és ellenőrzik az alkalmazás működését. A tesztek segítségével ellenőriztük az alkalmazás működését és funkcionalitását, valamint azonosítottuk az esetleges hibákat.

A végrehajtott tesztlépések a mellékelt "Teszt\_dokumentacio.xlsx" dokumentumban találhatók.

## 2.5. Fejlesztési lehetőségek

Számunkra nagyon fontos volt annak a lehetősége, hogy a későbbiekben a régi és új ügyfelek igényeire tudjuk szabni a programot, ezért fejlesztési lehetőségekkel is kalkuláltunk a jövőbeni rugalmasság érdekében.

Az egyik ilyen gondolatunk az online fizetés lehetőségének bevezetése volt mivel az online fizetés lehetővé teszi a vásárlók számára, hogy otthonról vagy bárhol máshonnan vásároljanak, amennyiben internetkapcsolat áll rendelkezésre. Nem szükséges az üzletekbe vagy bankfiókokba látogatniuk fizetés céljából. Ez a módszer általában gyorsabb, mint a hagyományos fizetési módoké. Ez időt takaríthat meg a vállalkozások számára. A fizetési folyamatok nyomon követhetőek és dokumentálhatóak.

A másik zászlóshajó a parkolási lehetőségek megjelenítése, mivel ez alapvetően egy kardinális kérdéskör minden nagyobb esemény idején, így a vásárok alkalmával is. A weboldalon meg lehetne tekinteni a parkolási zónák pontos elhelyezkedését és akár annak telítettségét is.

Amennyiben erre igény van megjelenítenénk a helyi látványosságok elérhetőségeit képekkel, az oldal jobb oldalán. Egymás alá helyeznénk ki a linkeket és arra kattintva eljutnánk a látnivaló weboldalára.

Későbbiekben szeretnénk az árusok listájában is keresési lehetőséget biztosítani a felhasználóinknak, hogy könnyebben fel tudják venni a kapcsolatot az eladókkal.

### 3. Felhasználói dokumentáció

#### 3.1. Szükséges hardver eszközök és szoftverek

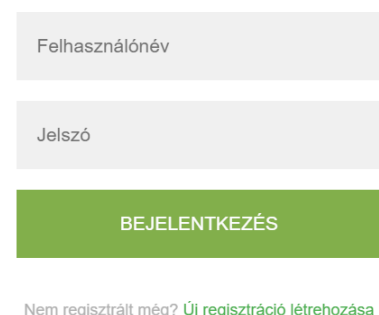
Az oldal funkcionális felépítése a PHP, HTML, CSS, JavaScript és AJAX technológiákra épül, amelyek biztosítják a weboldal gyors és megbízható működését minden mai böngészőben. Az adatkezelést MySQL adatbázis szolgálja, mely az adatok biztonságos tárolását és gyors elérhetőségét garantálja.

Az oldal kényelmesen használható egy mai böngésző futtatására alkalmas okos telefonról éppúgy, mint otthon a laptop és az asztali számítógépről. Javasolt böngészők: Mozilla Firefox, Google Chrome, Brave, Safari

#### 3.2. A program részletes bemutatása

##### 3.2.1. Bejelentkezés

A "Bejelentkezés" menüpont az oldalsó menüsorban található, ahol a felhasználók felhasználónév és jelszó megadásával tudnak belépni a rendszerbe, amely kifejezetten az árusok és adminisztrátorok számára lett kialakítva. Bejelentkezni csak azok a felhasználók jogosultak, akiket az adminisztrátor előzetesen jóváhagyott.



The screenshot shows a login interface with two input fields: 'Felhasználónév' (Username) and 'Jelszó' (Password). Below these fields is a green button labeled 'BEJELENTKEZÉS' (Login). At the bottom, there is a link that says 'Nem regisztrált még? Új regisztráció létrehozása' (Not registered yet? Create new registration).

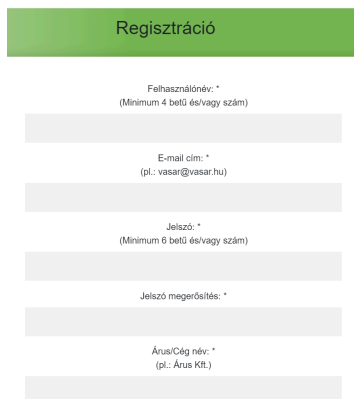
A bejelentkezési folyamat során a rendszer ellenőrzi a megadott felhasználónév és jelszó helyességét. Amennyiben a felhasználónév nem szerepel az adatbázisban, vagy a jelszó nem egyezik meg a tárolt hash értékkel, a rendszer hibaüzenetet jelenít meg. További ellenőrzések is történnek, például a fiók státuszának vizsgálata, mint például az, hogy jóváhagyásra vár-e, felfüggesztették-e vagy törölték-e.

Ezt követően a felhasználókat a főoldalra irányítja, ahol tájékoztatást kapnak a bejelentkezés sikerességéről vagy esetleges hibákról. Ez a folyamat biztosítja, hogy csak azonosított és engedélyezett felhasználók férhessenek hozzá a rendszerhez, miközben a felhasználói adatok védve vannak az adatbázisban tárolt titkosított jelszavak által.

### 3.2.2. Regisztráció

A regisztrációs weboldal célja az új felhasználók regisztrációja a rendszerbe, hogy hozzáférjenek a weboldal kínálta szolgáltatásokhoz és funkciókhoz. A weboldal lehetővé teszi a felhasználók számára, hogy létrehozzanak egy felhasználói fiókot, és megadják az alapvető információkat magukról és vállalkozásukról.

#### Regisztrációs Útmutató:



The image shows a registration form with a green header labeled 'Regisztráció'. Below the header, there are five input fields with labels and examples:

- Felhasználónév: \***  
(Minimum 4 betű és/vagy szám)
- E-mail cím: \***  
(pl.: vasar@vasar.hu)
- Jelszó: \***  
(Minimum 6 betű és/vagy szám)
- Jelszó megerősítés: \***
- Árus/Cég név: \***  
(pl.: Árus Kft.)

#### Felhasználónév:

- Adj meg egy felhasználónevet. Ez legalább egy karakterből kell álljon.
- A felhasználónév csak betűket, számokat és szóközöket tartalmazhat.
- A felhasználónév hossza legyen legalább 4 karakter, de legfeljebb 20 karakter.

#### Email cím:

- Adj meg egy érvényes email címet.
- Az email cím formátumának megfelelőnek kell lennie (pl. valami@example.com).

#### Jelszó:

- Adj meg egy jelszót.
- A jelszónak legalább egy karakterből kell állnia.
- A jelszó csak betűket, számokat és szóközöket tartalmazhat.
- A jelszónak legalább 6 karakter hosszúnak kell lennie, de legfeljebb 50 karakter hosszú.

#### Jelszó megerősítés:

- Írd be újra a jelszavad a megerősítés céljából.

- A jelszó megerősítése mezőben pontosan ugyanazt a jelszót kell megadnod, mint amit az előző mezőben adtál meg.

Cég neve:

- Add meg a cég nevét.
- A cég nevének legalább 5 karakter hosszúnak kell lennie, de legfeljebb 50 karakter hosszú.

Árus/Cég név: \*  
(pl.: Árus Kft.)

Kapcsolattartó:

Telefonszám: \*  
06701111333

Online elérhetőség:  
(pl.: <https://www.facebook.com/>)

Termék leírás \*  
(pl.: Termékünkben többféle méz megtalálható, többek között akác, repce, vegyes virágméz)

Termék kategória: \*

☐ Állateledél
 ☐ Ásvány
 ☐ Babaholmi
 ☐ Babaruhák
 ☐ Bőr
 ☐ Bőrúru
 ☐ Cipő
 ☐ Édesség

Kapcsolattartó:

- Add meg a kapcsolattartó nevét.
- A kapcsolattartó neve csak betűket, számokat és szóközőket tartalmazhat.

Telefonszám:

- Add meg a telefonszámod.
- A telefonszámnak 11 számjegyből kell állnia, és az első két számjegynek 06-nak kell lennie.

Online elérhetőség (opcionális):

- Ha rendelkezzel online elérhetőséggel, akkor add meg annak a webcímét.
- Az online elérhetőségnek érvényes URL címnek kell lennie.

Termék leírás:

- Adj meg egy rövid leírást a termékedről.
- A termék leírása legalább 3 karakter hosszúnak kell lennie, de legfeljebb 50 karakter hosszú.

Termék kategória:

- Válassz legalább egy termék kategóriát a legördülő menüből.



Kérlek, ügyelj arra, hogy minden mezőt helyesen tölts ki, és kövesd az adott instrukciókat az űrlap kitöltése során. Ha bármilyen kérdésed vagy problémád merülne fel a regisztráció során, ne habozz felvenni a kapcsolatot velünk!

### 3.2.3. Profil

**Személyes adatok**

Árus/Cég neve: \*

Profil Kft.

E-mail cím: \*

brumm@gmail.com

Kapcsolattartó:

Jónás Patrícia

Telefonszám: \*  
(pl.: 06301234567)

06705536254

Online elérhetőség:  
(pl.: https://www.pelda.com)

A felhasználói profil oldalunk célja, hogy a felhasználók személyes és felhasználói adataikat megtekinthessék és módosíthassák. A weboldal lehetővé teszi a felhasználók számára, hogy frissítsék a regisztráció során megadott adatokat, valamint hogy változtassanak jelszavukon.

https://www.brummkft.hu

Bemutakozás: \*

Termékinálatunkban többféle méz megtalálható, többek között akác, repce, vegyes virágméz.

Termékkategória: \*

Állateledel ☒ Ásvány ☐ Babaholmi ☒ Babaruha ☒ Bőr ☐ Bőraru ☐ Cipő ☐ Édesség ☐

Ékszer ☐ Élelmiszer ☐ Faipari termék ☐ Fűszer ☐ Gyógynövény ☐ Gyümölcs ☐ Hal ☐

Hús ☐ Játék ☐ Kávé ☐ Kerámia ☐ Kerti szerszám ☐ Kézműves szappan ☐

Kézműves tárgy ☐ Méz ☐ Népi kézműves termék ☐ Növény ☐ Óra ☐ Pálinka ☐

Papucs ☐ Régiség ☐ Ruhanemű ☐ Sajt ☐ Szendvics ☐ Szobanövény ☐ Szőnyeg ☐

Táska ☐ Tea ☐ Tűzzománc ☐ Vetőmag ☐ Virág ☐ Zöldség ☐

MÓDOSÍT

**Felhasználói adatok**

Felhasználónév: \*

brumm1

Jelenlegi jelszó: \*

Új jelszó: \*  
(Minimum 6 betű és/vagy szám)

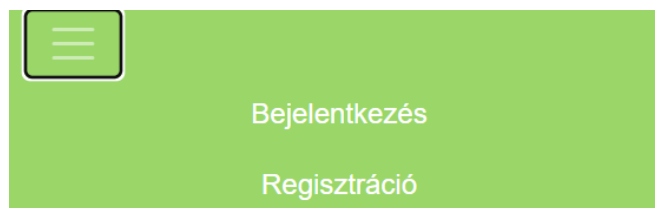
Jelszó megerősítése: \*

MÓDOSÍT

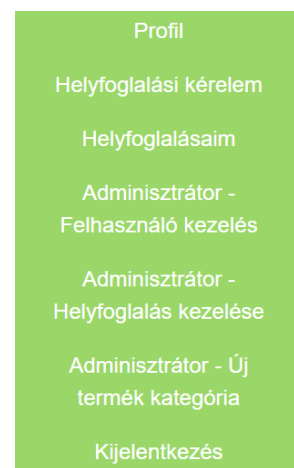
Külön lehetőséget adunk arra is, hogy termékkategóriát lehessen bővíteni vagy épp szűkíteni.

#### 3.2.4. Oldalsó menü

A navigációs sáv egy összeecsukható menüt tartalmaz, amely a kisebb képernyőkön egy kapcsoló gombra kattintva bontható ki. A menü tartalma a felhasználó bejelentkezési állapotától függ. Amennyiben a felhasználó nincs bejelentkezve, a menü a "Bejelentkezés" és a "Regisztráció" opciókat jeleníti meg, lehetőséget biztosítva az oldalra történő bejelentkezésre vagy új felhasználói fiók létrehozására.



Ha a felhasználó már bejelentkezett, számos további menüpont válik elérhetővé, beleértve a "Profil", "Helyfoglalási kérelem" és "Helyfoglalásaim" opciókat. Amennyiben a felhasználó adminisztrátori jogokkal rendelkezik, további lehetőségek, mint a "Felhasználó kezelés", "Helyfoglalás kezelése" és "Új termék kategória" is megjelennek. Ezek a menüpontok lehetővé teszik az adminisztrátorok számára a felhasználók, a helyfoglalások és az új termékkategóriák kezelését.



A "Kijelentkezés" menüpont lehetőséget biztosít a bejelentkezett felhasználóknak a rendszerből való kijelentkezésre, ezzel megszakítva a munkamenetet és eltávolítva a felhasználói adatokat a sütikből és a munkamenetből.

#### 3.2.5. Keresés

A keresés az oldal fejlécében található, mely minden oldalon megjelenik és minden látogató számára elérhető funkció.

A keresés funkcióval termékkategóriákra kereshetünk, melyet a kereső mezőbe írt placeholder mutat a felhasználó számára.

Miután beírtuk a keresett szót és rákattintottunk a keresés gombra, a rendszer két lehetőséget kínál nekünk:

Ha a keresett szóra nincs találat, egy üzenet jelenik meg: "Nincs találat a(z) 'xy' termékkategóriára!". Ez segít abban, hogy tudjuk, a keresésünk eredménytelen volt, és továbbléphetünk más lehetőségek felé.

A következő táblázatokat láthatjuk, amennyiben a keresés sikeres és van találat:

- a jövőbeli helyfoglalások táblázata azt mutatja, hogy mely eladók foglaltak helyet a jövőbeni vásárokon, így megtudhatjuk melyek azok a vásárok, amelyeken elérhető lesz a keresett termék
- a múltbeli helyfoglalások táblázata azt mutatja, hogy mely eladók árusították a már a múltban vásáron a keresett terméket, így korábbi tapasztalatok alapján is tájékozódhatunk.

A keresés eredménye a foglalt hely számát is megjeleníti, így a vásárló tájékozódhat, hogy az elmúlt vásárokon hol találkozhatott az eladóval, valamint hogy a jövőbeli vásárokon hol találja meg pontosan. A tájékozódásban a táblázatok alatt megjelenő vásár térkép ad segítséget.

Találatok "méz" termékkategóriára a következő vásárokon:

Termék kategória	Árus	Vásár időpontja	Foglalt hely száma
Méz	Brumm Kft	2024-05-19	I/1

Találatok "méz" termékkategóriára az elmúlt vásárokon:

Termék kategória	Árus	Vásár időpontja	Foglalt hely száma
Méz	Brumm Kft	2024-04-21	I/1



### 3.2.6. Árusok listája

Az "Árusok listája" egy nyilvánosan elérhető menüpont, ahol bárki számára lehetőség nyílik a kiállítók bemutatkozásának, és elérhetőségeinek böngészésére. Amint erre a menüpontra kattint, egy táblázat jelenik meg, amely ABC rendben sorolja fel az adminisztrátor által jóváhagyott regisztrált felhasználókat.

Ez a lista segít abban, hogy könnyen áttekinthető legyen, mely kiállító található meg az eseményen, milyen termékkínálattal rendelkezik, valamint hogyan lehet őt elérni. Ez által a látogatónak lehetősége van tájékozódni, hogy melyik standot szeretné felkeresni, és hogy az adott kiállítónál milyen áruk érhető el.

Árus neve	Tevékenységi kör	Telefonszám	Online elérhetőség
AranyKéz	Kézműves ékszerek különleges alkotások és egyedi darabok	06303456789	<a href="https://www.facebook.com/AranyKézkovacspeter1@example.com">https://www.facebook.com/AranyKézkovacspeter1@example.com</a>
Brumm Kft	Termékkínálatunkban többféle méz megtalálható, többek között akác, repce, vegyes virágméz.	06705536254	<a href="https://www.brummkft.hu">https://www.brummkft.hu</a> brumm@gmail.com
EcoBioTermék	Biogazdálkodásból származó termékek, organikus konyhai alapanyagok és higiéniai termékek	06207654321	<a href="https://www.facebook.com/EcoBioTermék">https://www.facebook.com/EcoBioTermék</a> nagymaria2@example.com
GyógyFűszerek	Gyógyító hatású fűszerek, teakeverékek és gyógynövények, természetes gyógymódok	06705646456	<a href="https://www.facebook.com/GyógyFűszerek">https://www.facebook.com/GyógyFűszerek</a> molnaristvan7@example.com
KerámiaMania	Kézzel készített kerámia díszek, egyedi cseréptárgyak és dekorációs elemek	06309789789	<a href="https://www.facebook.com/KerámiaMania">https://www.facebook.com/KerámiaMania</a> feherkata8@example.com
MűvészKuckó	Festmények, művészeti alkotások és díszítőelemek, egyedi ajándéktárgyak	06707888999	<a href="https://www.facebook.com/MűvészKuckó">https://www.facebook.com/MűvészKuckó</a> szabolaura5@example.com

### 3.2.7. Helyfoglalási kérelem

A "Helyfoglalási kérelem" menüpont lehetőséget biztosít a regisztrált felhasználóknak, hogy helyet foglalhassanak az elkövetkező vásár időpontokra. Egyetlen kattintásra van szükség az oldalsó menüben található hivatkozásra, hogy megnyíljon az oldal, ahol egy legördülő listából választhatják ki a kívánt időpontot, amelyre szeretnének helyet foglalni. Emellett egy térkép is rendelkezésre áll, ami segíti a felhasználókat abban, hogy könnyen megtalálják a számukra megfelelő árusító helyet.

Kérem válassza ki, melyik időpontban szeretne helyet foglalni!

2024-07-21 ▾

Ebben az időpontban a következő helyek szabadok:

VIII/5 ▾

Kiválasztom

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.
I.	■	■	■	■	■	■	■	■	■	■	■	■
II.	■	■	■	■	■	■	■	■	■	■	■	■
III.	■	■	■	■	■	■	■	■	■	■	■	■
IV.	■	■	■	■	■	■	■	■	■	■	■	■
V.	■	■	■	■	■	■	■	■	■	■	■	■
VI.	■	■	■	■	■	■	■	■	■	■	■	■
VII.	■	■	■	■	■	■	■	■	■	■	■	■
VIII.	■	■	■	■	■	■	■	■	■	■	■	■

Vásár térkép

BEJÁRAT

Regisztráció

Miután a felhasználó kiválasztotta a kívánt dátumot, az oldal megjeleníti a szabad helyek listáját az adott időpontban. A felhasználó választhat a listából, majd megjelenik a "Kiválasztom" gomb. Ezen gombra kattintva az oldal átirányít a foglalás véglegesítésére szolgáló oldalra, ahol összefoglalásra kerülnek a foglalás adatai, beleértve a foglaló adatait, valamint a foglalt hely részleteit.

### A kiválasztott hely foglalásának véglegesítése

**Foglaló adatai:**

Cégnév: MűvészKuckó  
 Kapcsolattartó: Szabó Laura  
 Telefonszám: 06707888999  
 E-mail cím: szabolaura5@example.com

**Foglalás adatai:**

Vásár dátuma: 2024-07-21  
 Hely száma: VIII/5  
 Hely ára: 1500 Ft

Foglalás véglegesítése

Az oldal alján található "Foglalás véglegesítése" gombra kattintva az oldal visszairányít az index oldalra, és egy felugró ablakban értesülhetünk arról, hogy a foglalás sikeres volt-e. Sikeres helyfoglalás esetén a kérelem adminisztrátori jóváhagyásra vár, és helyfoglalás menüpontban tekinthető meg az aktuális állapota (jóváhagyásra vár, jóváhagyott vagy törölt formájában). Ezáltal a felhasználó teljes körű tájékoztatást kap a foglalás állapotáról, és könnyen nyomon követheti azt.

### 3.2.8. Helyfoglalásaim

A "Helyfoglalásaim" menüpont lehetőséget nyújt a regisztrált felhasználóknak arra, hogy tájékozódjanak helyfoglalásaik állapotáról. A helyfoglalások három különböző állapotban lehetnek: jóváhagyásra váró, jóváhagyott és törölt. Az oldal három külön táblázatban jeleníti meg ezeket, így könnyen áttekinthetően látható, hogy mely helyfoglalások várnak még

**Jóváhagyásra váró helyfoglalási kérelmek:**

Vásár dátuma	Foglalt hely sorszáma
2024-06-16	VIII/9

**Jóváhagyott helyfoglalási kérelmek:**

Vásár dátuma	Foglalt hely sorszáma
2024-05-19	I/1
2024-04-21	I/1

**Nincs törölt helyfoglalási kérelem!**

jóváhagyásra, melyek lettek már jóváhagyva, és melyeket töröltek.

Ez lehetővé teszi a felhasználók számára, hogy nyomon kövessék helyfoglalásaik státuszát és szükség esetén további intézkedéseket tegyenek.

### 3.2.9. Adminisztrátor - Felhasználó kezelés

A moderátorok itt tudják kezelni a felhasználói profilok státuszait. A rendszeren belül képesek megtekinteni és módosítani a felhasználók státuszát. Ez magában foglalja új jelentkezések aktiválását, meglévő felhasználók felfüggesztését vagy profilok törlését.

### Felhasználói státuszok kezelése

#### Új felhasználói jelentkezések

Aktiválás

Felhasználó név	Eladó/Cég név
<input type="radio"/> tothistvan	Gazdakör
<input type="radio"/> horvathandrea	BőrVilág
<input type="radio"/> bakosbalazs	PálinkaPince

A rendszerben a felhasználói információk egy táblázatban jelennek meg, amely tartalmazza a felhasználó nevet, a cég/árus nevét, és az aktuális státuszt. Amennyiben az egyik status kategória üres, a rendszer tájékoztatja az adminisztrátort, hogy nincsenek megjeleníthető adatok.

Aktív felhasználók

Felfüggeszt Töröl

Felhasználó név	Eladó/Cég név
<input type="radio"/> brumm1	Brumm Kft
<input type="radio"/> kovacspeter	AranyKéz
<input type="radio"/> nagymaria	EcoBioTermék
<input type="radio"/> kissgeza	ZöldKert
<input type="radio"/> szabolaura	MűvészKuckó
<input type="radio"/> molnaristvan	GyógyFűszerek
<input type="radio"/> feherkata	KerámiaMania

Felfüggesztett felhasználók

Aktiválás Töröl

Nincs felfüggesztett felhasználó!

Törölt felhasználók

Felhasználó név	Eladó/Cég név
galreka	TextilTrend

Az adminisztrátorok rádiógombok segítségével módosíthatják a felhasználói státuszokat. Minden művelet végrehajtása előtt a rendszer ellenőrzi, hogy a kiválasztott felhasználó azonosítója ki van-e töltve, ami megakadályozza az érvénytelen műveleteket.

A rendszer biztonsági funkciói közé tartozik, hogy csak az adminisztrátori jogosultsággal rendelkező felhasználók férhetnek hozzá. Azok, akik nem rendelkeznek ezzel a jogosultsággal, automatikusan az index oldalra irányítódnak át.

Amennyiben hiba lép fel, például adatbázis probléma vagy a felhasználói azonosító hiánya miatt, a rendszer hibaüzenetet jelenít meg, mely segít az adminisztrátoroknak a probléma gyors azonosításában és kezelésében.

### 3.2.10. Adminisztrátor - Helyfoglalás kezelése

Itt tekinthetik meg az adminok a függőben lévő, jóváhagyott és elutasított helyfoglalásokat, és dönthetnek a kérelmek sorsáról. A felület könnyen áttekinthető és egyszerűen kezelhető, ami megkönnyíti a különböző helyfoglalások állapotának gyors ellenőrzését és kezelését.

Az adminisztrátorok képesek módosítani a helyfoglalások státuszát egy egyszerű rádiógomb segítségével, amely lehetővé teszi a jóváhagyást vagy elutasítást. Ha a



moderátor nem jelölt ki rádiógombot, a rendszer figyelmezteti, hogy válasszon ki egy kérelmet a módosítás előtt.

### Helyfoglalási kérelmek kezelése

#### Új helyfoglalási kérelmek

Felhasználó név	A vásár dátuma	Az elárusító hely száma
<input type="radio"/> kissgeza	2024-04-21	III/3
<input type="radio"/> horvathandrea	2024-04-21	III/12
<input type="radio"/> feherkata	2024-04-21	IV/6
<input type="radio"/> galreka	2024-04-21	V/10
<input type="radio"/> horvathandrea	2024-04-21	III/12
<input type="radio"/> galreka	2024-04-21	V/10
<input type="radio"/> kovacspeter	2024-07-21	I/2
<input type="radio"/> feherkata	2024-08-18	IV/6

A helyfoglalási kérelmek táblázatos formában jelennek meg, ahol az adminisztrátorok láthatják a felhasználók nevét, a vásárok dátumait és a helyek számát. Ha nincsenek függőben lévő kérelmek, a rendszer tájékoztatja az adminisztrátort, hogy jelenleg nincs elbírálásra váró kérelem.

#### 3.2.11. Adminisztrátor - Új termék kategória

Az adminisztrátorok ezen az oldalon tudnak új termékkategóriákat felvenni, mely két alapvető funkcionalitást biztosít: új kategóriák hozzáadása az adatbázishoz és a meglévő kategóriák megjelenítése egy táblázat formájában. Ez az egész folyamat egy biztonságos munkameneten alapszik, amely biztosítja, hogy csak a megfelelő jogosultságokkal rendelkező, bejelentkezett felhasználók érhessék el ezeket a funkciókat.

### Új termékkategóriák felvétele

**Termék kategória:**

☐ Állateledel

☐ Ásvány

☐ Babaholmi

Amikor az adminisztrátor a termékkategória hozzáadására szolgáló mezőt kitölti és elküldi, a kód ellenőrzi, hogy a megadott kategória név üres-e. Ha igen, a rendszer egy hibaüzenetet jelenít meg, amely felszólítja a felhasználót, hogy adjon meg egy érvényes kategória nevet. Ha a megadott név érvényes, az új kategória bekerül az adatbázisba.

Az űrlap alatt egy táblázat jelenik meg, amely az összes jelenlegi termékkategóriát listázza. Ez a táblázat segít az adminisztrátornak abban, hogy áttekintse a már létező kategóriákat, így könnyebben dönthet arról, hogy szükséges-e új kategóriát hozzáadni. Ha probléma merül fel a kategóriák lekérdezése közben, a rendszer egy hibaüzenetet jelenít meg, tájékoztatva az adminisztrátort a fellépő problémáról.

### 3.2.12. Adminisztrátor - A következő vásár beállítása

## VÁSÁR

A következő vásár időpontja:

2024-05-19

[Szerkesztés](#)

Ez a felugró ablak az adminisztrátoroknak nyújt segítséget a vásárok dátumainak hatékony kezeléséhez, lehetővé téve új időpontok hozzáadását, a következő esemény kiválasztását és az aktuális vásári időpont megtekintését az

adatbázison keresztül. A kezdőlapon az adminisztrátorok a "Szerkesztés" linkre kattintva egy felugró ablakon, legördülő menü segítségével könnyedén elérhetik és kezelhetik a vásári dátumokat, míg egy egyszerű kezelői interface-en keresztül frissíthetik a kiválasztott eseményeket.

#### A következő vásár időpontja legyen:

Új vásár dátum hozzáadása:

Jelölje ki a következő

Kiválasztás

2024. április

H	K	Sze	Cs	P	Szo	V
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

TörölésMa

A rendszer hibakezelése biztosítja, hogy hiányos vagy helytelen felhasználói adatok esetén a felület hibaüzenetet jelenítsen meg. Az adatbázis-műveletek során felmerülő problémák részletes jelentése segít az adminisztrátoroknak a hibák gyors azonosításában és kezelésében.

### 3.3. Helytelen használatból adódó hibajelzések magyarázata

Amennyiben a felhasználó szeretné az oldalon lévő funkciókat használni, úgy megeshet, hogy bizonyos felületeken találkozik használatból adódó hiba jelzésekkel.

Űrlap kitöltése során a következő esetekben találkozhatunk üzenetekkel:

#### 3.3.1. Üres mezők hibái:

Felhasználónév, email, jelszó, jelszó megerősítése, termék leírás és cég név mezők kitöltése kötelező. Ha ezek valamelyike üres marad, az egy vagy több hibaüzenetet eredményez.

#### 3.3.2. Formátum ellenőrzések:

Az email cím formátumát ellenőrzi a JavaScript és a PHP is. Ha az email cím nem megfelelő formátumú, hibaüzenet jelenik meg.

A jelszó és a jelszó megerősítése közötti egyezőség ellenőrzése. Ha nem egyeznek, hibaüzenet jelenik meg.

A telefonszám formátumát ellenőrzi a PHP. Ha az nem megfelelő, hibaüzenet jelenik meg.

Az online elérhetőség esetén URL formátumot vár, és ha nem megfelelő, hibaüzenet jelenik meg.

#### 3.3.3. Karakter hossz ellenőrzések

A felhasználónév, jelszó, cégnév és termékleírás mezők karakterhosszát korlátozza a PHP. Ha ezek a mezők nem felelnek meg a megadott hosszúságnak, hibaüzenet jelenik meg.

Adatbázis ellenőrzések:

Ellenőrzi, hogy a felhasználónév már létezik-e az adatbázisban. Ha igen, hibaüzenet jelenik meg.

Ellenőrzi, hogy az email cím már létezik-e az adatbázisban. Ha igen, hibaüzenet jelenik meg.

#### 3.3.4. Termék kategória ellenőrzés

A termék kategória választásának hiányát ellenőrzi a PHP. Ha nincs választás, hibaüzenet jelenik meg.

#### 3.3.5. Általános hibák

Általános hibák esetén a hibaüzenetek vörös színnel jelennek meg, hogy könnyen észrevehetőek legyenek a felhasználó számára.

Ezek az ellenőrzések és hibaüzenetek segítenek a felhasználónak helyesen kitölteni és elküldeni az űrlapot.

#### 3.4. Információkérés lehetőségeinek megadása

A projektről további információ az alábbi e-mail címen kérhető: [hellovasar.info@gmail.com](mailto:hellovasar.info@gmail.com)

### 4. Összefoglalás

A projekt egy online platform létrehozását célozta meg, amely modern digitális technológiákat integrál a hagyományos vásári környezetben, javítva ezzel az árusok és vásárlók közötti interakciókat és élményt. Kezdeti célkitűzéseinket a projekt keretein belül sikeresen megvalósítottuk, ugyanakkor arra is rá kellett jönnünk, hogy olyan ez, mint a házépítés, mindig felmerül legalább egy apróság, amivel még jobba, még teljesebbé lehet tenni a dolgokat. Több olyan bővítési lehetőség is felmerült bennünk, amiket a korlátozott időkeret miatt nem állt módunkban megvalósítani, ugyanakkor reméljük, hogy a jövőben már ezek megvalósítási problémáin törhetjük a fejünket. Bízunk benne, hogy pár év múlva, ha a vásárban sétálunk a mi weboldalunk segítségével már könnyedén megtalálhatóak lesznek a kiszemelt különlegességeket

áruló kézművesek épp úgy, mint a néni, aki a legfinomabb epret árulja vagy a bácsi akitől mindig be lehet szerezni egy szép, régi dísz tárgyat vagy egy könyvet, aminek mi a sokadik lelkes tulajdonosai leszünk, de akár egy frissen kiötlött ajándék megtalálásában is jó szolgálatot tehet.

Ez a hibrid megközelítés nem csupán újít a piaci élményben, hanem a jelenkor fogyasztói és eladói szokásaira is reflektál, szélesebb közönség számára téve elérhetővé a hagyományos piaci élményt és növelve a kereskedelmi és kulturális interakciókat.

## 5. Köszönetnyilvánítás

Szeretnénk megköszönni a képzési lehetőséget a Szegedi SZC Tóth János Szakképző Iskolának és Szilágyi Mihály Kollégiumnak, valamint a Mórahalmi Ágazati Képző Központ Nonprofit Kft.-nek. Különösen Némethné Torma Idának, aki mindig türelmesen válaszolt kérdéseinkre. Hálásak vagyunk tanárainknak, Tóth Attilának és Árgilán Viktornak, valamint osztályfőnökünknek, Szabó Beatrixnek, Szabó Gábornak és Balog Attilának, akik rengeteg segítséget és türelmet tanúsítottak irántunk a nehéz időkben. Csatatunk, mely anyukákból áll, számos kihívással nézett szembe, de tanáraink és a tagok közötti kölcsönös támogatás lehetővé tette, hogy rugalmasan kezeljük az akadályokat és sikeresen túl jussunk a nehézségeken. Nem utolsó sorban szeretnénk megemlíteni családtagjainkat, akiknek odaadó támogatása nélkül nem sikerült volna elkészítenünk a projektet és akik szó nélkül viselték, hogy anyu már megint a helyi pizzériából főzött.

## 6. Irodalomjegyzék

<https://www.w3schools.com/php/default.asp>

<https://getbootstrap.com/docs/4.6/getting-started/introduction/>