

ZÁRÓDOLGOZAT

Dreilinger Vanessza Maja

Családi Alexandra

2024

Kertészeti Webáruház

Dreilinger Vanessza Maja

Családi Alexandra

SZÁMALK-Szalézi Technikum és

Szakgimnázium

Szoftverfejlesztő

Nyilatkozat

a záródolgozat eredetiségéről

Alulírott Dreilinger Vanessza Maja (név) {Fehér Andrea. (anya neve) 732403NE. (szem. ig. szám)} büntetőjogi és fegyelmi felelősségem tudatában kijelentem és aláírással igazolom, hogy a záródolgozat saját munkám eredménye. A felhasznált irodalmi és egyéb információs forrásokat az előírásoknak megfelelően kezeltem, a záródolgozat készítésre vonatkozó szabályokat betartottam.

Kijelentem, hogy ahol mások eredményeit, szavait vagy gondolatait idéztem, azt a záródolgozatomban minden esetben, beazonosítható módon feltüntettem, a dolgozatban található fotók és ábrák közlésével pedig mások szerzői jogait nem sértem.

Kijelentem, hogy a záródolgozatom elektronikus változata teljes egészében megegyezik a nyomtatott formával.

Hozzájárulok ahhoz, hogy az érvényben lévő jogszabályok és a Számalk-Szalézi Szakgimnázium belső szabályzata alapján az iskola saját könyvtárában megtekinthető (olvasható) legyen a záródolgozatom.

A záródolgozat titkos/nem titkos.

Budapest, 2024. május 2

Dreilinger Vanessza Maja
Tanuló aláírása

Nyilatkozat

a záródolgozat eredetiségéről

Alulírott Családi Alexandra (név) {Bálint Tímea. (anya neve) 428080EE. (szem. ig. szám)} büntetőjogi és fegyelmi felelősségem tudatában kijelentem és aláírással igazolom, hogy a záródolgozat saját munkám eredménye. A felhasznált irodalmi és egyéb információs forrásokat az előírásoknak megfelelően kezeltem, a záródolgozat készítésre vonatkozó szabályokat betartottam.

Kijelentem, hogy ahol mások eredményeit, szavait vagy gondolatait idéztem, azt a záródolgozatomban minden esetben, beazonosítható módon feltüntettem, a dolgozatban található fotók és ábrák közzétételével pedig mások szerzői jogait nem sértem.

Kijelentem, hogy a záródolgozatom elektronikus változata teljes egészében megegyezik a nyomtatott formával.

Hozzájárulok ahhoz, hogy az érvényben lévő jogszabályok és a Számalk-Szalézi Szakgimnázium belső szabályzata alapján az iskola saját könyvtárában megtekinthető (olvasható) legyen a záródolgozatom.

A záródolgozat titkos/nem titkos.

Budapest, 2024. május .



Tanuló aláírása

Tartalom

1	Programhoz szükséges adatok	7
1.1	Github linkek	7
1.2	Felhasználó nevek és jelszavak	7
2	Bevezetés	8
2.1	Témaválasztás	8
3	Tervezés, drótvázak	9
3.1	Tervezés folyamata	9
3.2	Drótvázak	10
4	Felhasználói dokumentáció	14
4.1	A program célja és lényegesebb funkciói	14
4.2	Futtatási környezet	14
4.3	A program fő funkcióinak leírása	14
4.3.1	A vásárló funkciói	14
4.3.2	Az admin funkciói:	15
5	Fejlesztői dokumentáció	17
5.1	Fejlesztői környezet	17
5.2	Program struktúra	17
5.3	Adatbázis	18
5.3.1	Egyed-kapcsolati diagram	18
5.3.2	Az adatrendszer ismertetése	18
5.3.3	Az ügyviteli funkciók	19
5.3.4	Az adatbázis-tervezés a szükséges egyedek kialakításával	19
5.3.5	Táblák	22
5.3.6	Kapcsolati ábra	25
5.3.7	Hasznos Lekérdezések	25
5.3.8	Kiemelt hasznos lekérdezések	26
5.3.9	Beépítendő megszorítások	27
5.3.10	Technikai táblák (nem kötelezően megvalósítandó)	28

5.4	Backend	28
5.4.1	Modellek és Controllerek.....	28
5.4.2	API végpontok.....	33
5.5	Frontend	36
5.6	Reszponzivitás	36
6	Tesztelés	45
7	Fejlesztési lehetőségek	50
8	Irodalomjegyzék	1

1 Programhoz szükséges adatok

1.1 Github linkek

https://github.com/Kerteszek/Kerteszeti_Frontend.git

https://github.com/Kerteszek/Kerteszeti_Backend.git

1.2 Felhasználó nevek és jelszavak

Felhasználó próba adatokkal

belus@gmail.com

Aa123@

Admin

admin@gmail.com

Aa123@

SuperAdmin

sadmin@gmail.com

Aa123@

2 Bevezetés

A Szakdolgozat témája egy elképzelt Kertészeti webáruház, ahol az elérhető növényeket lehet megtekinteni és megvásárolni.

Az oldalon a vásárlás **regisztrációval** történik, enélkül nem lehet vásárolni.

A regisztráláshoz kell megadni az *email címet* és *nevet*.

A vásárláshoz meg kell adni a felhasználó nevét, számlázási és rendelési címét.

Lehetőség van az adatok módosításra is, ha a szállítási adatok változnak

Admin-ként az admin felületen képes új termékek felvételére, azok árának változtatására, és új készlet beszerzésére adott termékből. Ki tudja listázni az összes terméket.

A SuperAdmin kezeli a felhasználókat, **törölheti** őket, továbbá adhat **admin jogosultságot**. Mindenre képes, mint amire az admin is.

2.1 Témaválasztás

A témaválasztás a következők alapján történt. Közös kerestünk számunkra is megfelelő és érdekes témát. Beszélgettünk különböző közös érdeklődési körökről, mi az amit szívesen látnánk viszont. Akadt pár téma, amit szívesen csináltunk volna: virágok, kertészeti cikkek. Végül ezeknek az ötvözetéből készült el a kertészeti webáruház. A végső döntést mellett szöveg, hogy sok kertészeti cég weboldala nem megfelelő a felhasználó szempontjából. Például a kényelmetlen elrendezések, a kategorizálások furcsasága. Ezek miatt szerettünk volna egy szép és kényelmes, felhasználóbarát webáruházat létrehozni. Illetve az adatbázis nagysága is döntőérv volt, az elejétől kezdve ötleteltünk, a későbbiekben ez csak bővült.

3 Tervezés, drótvázak

3.1 Tervezés folyamata

Témaválasztás

A tervezés egy rendkívül fontos folyamat bármely alkalmazás elkészítéséhez. A mi programunk elkészítésének legelső lépése a témaválasztás volt. Ezután összeszedtük a legfontosabb alap funkcióit készülendő alkalmazásunkhoz. Itt fontos volt, hogy megvalósítható célokat tűzzünk ki. Ezek után létrehoztunk kettő darab GITHUB projectet, hogy számon tudjuk követni a majd elvégzendő feladatainkat. Ezek a Kertészeti Backend, és Kertészeti Frontend elnevezést viselik.

Adatbázis tervezés

A következő nagy lépés az adatbázis szerkezetének megtervezése következett. Először is a legfontosabb adattáblákat kellett meghatározni, ezeknek ki kellett szolgálnia a már megtervezett funkcióink szükségleteit. Egy másik fontos lépéssel párhuzamosan a táblák egymással való kapcsolatának kialakítása volt. Továbbá eldöntöttük, hogy milyen adatokat szeretnénk ezekben eltárolni, illetve, hogy ezek milyen típusúak.

Természetesen ezeket később normalizáltuk, mivel a tervezés fázisában még nem voltak a legmegfelelőbbek egy élő adatbázishoz. Ezen lépés után elkészítettünk hasznos segéd, és szótártáblákat.

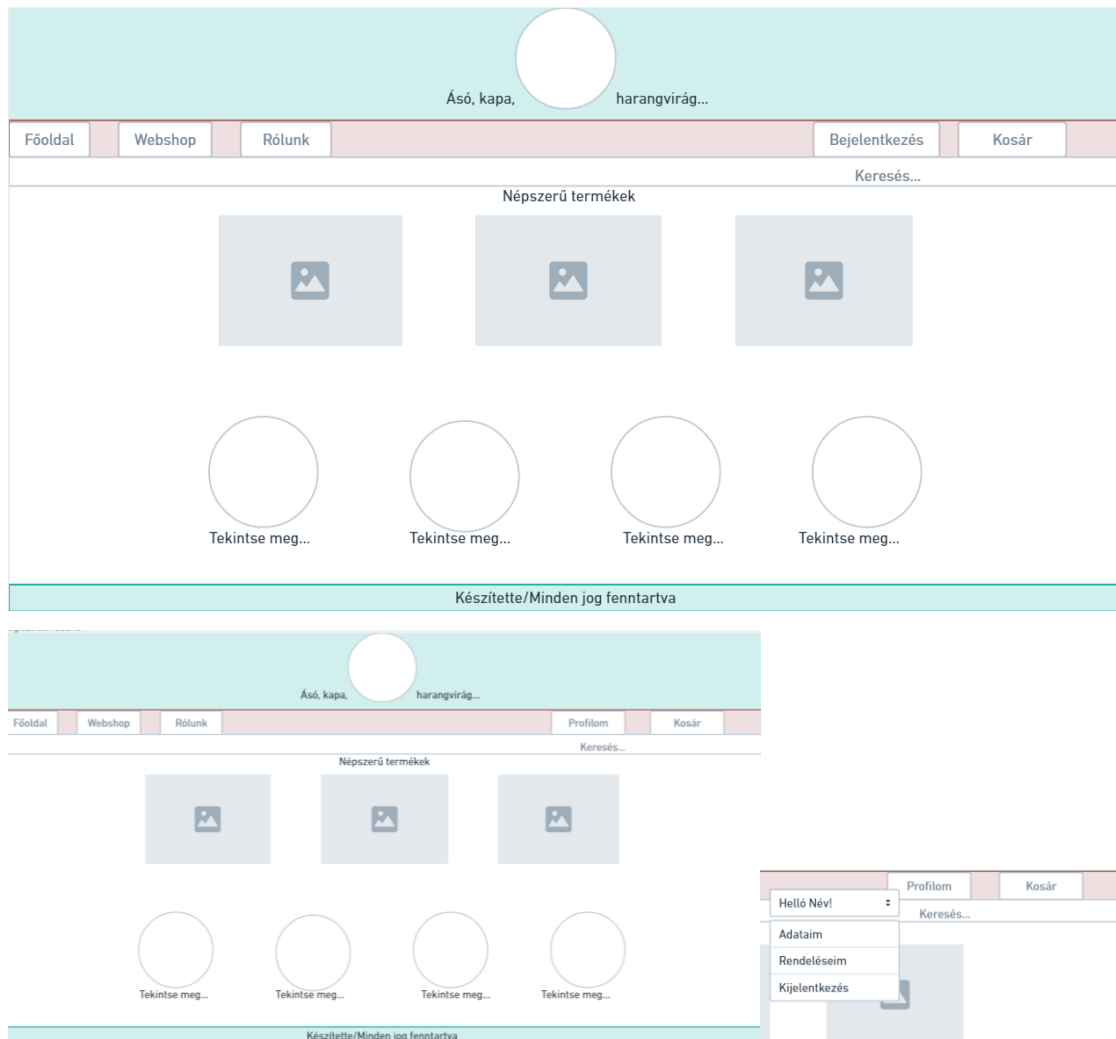
Látvány tervezés

A Whimsical nevű online oldalt használtuk a drótvázak tervezéséhez. Itt először is a főoldal megtervezésével kezdtük, hiszen ez az az oldaldal, amit bármely jövőbeli felhasználó legelőször fog látni. A következő a termékeket kilistázó felület volt a konkrét termék oldalával együtt. A tervek között volt a bejelentkezés és a regisztráció oldalainak megtervezése, ezek hasonló sémára épülnek, nagyon kicsi eltérés van közöttük. A látvány tervezés legvégén elkészítettük az admin oldalak kezdetleges drótvázát.

A tervezés közben igyekeztünk egy könnyen kezelhető felhasználó barát oldalt kialakítani, amit intuitívan lehet majd használni. Az

3.2 Drótvázak

Főoldal




Termékek oldal



Termék oldal



Admin és SuperAdmin oldalak

SzuperAdmin 

kijelentkezés

Készlet

Termékek

Rendelés


Felhasználók

Árazás

Kedvezmények

Jogosultságok

Helló, SzuperAdmin!

Admin 

kijelentkezés

Készlet

Termékek

Rendelés

Árazás

Kedvezmények

Helló, Admin!

Bejelentkezés

Ásó, kapa,harangvirág...

FőoldalWebshopRólunkBejelentkezésKosár

Keresés...

Bejelentkezés

Email

Jelszó

Bejelentkezés

Még nincs regisztrálva?
Regisztráció

Készítette/Minden jog fenntartva

Regisztráció

Ásó, kapa,harangvirág...

FőoldalWebshopRólunkBejelentkezésKosár

Keresés...

Regisztráció

Név

Email

Jelszó

Jelszó újra

Regisztráció

Készítette/Minden jog fenntartva

4 Felhasználói dokumentáció

A felhasználó meglátogatja a weboldalt. Az üdvözlőoldal után a legördülő menüből választhat tetszőleges menüpontot.

A főoldal menüpont minden esetben visszavisz a kezdőlapra. Válassza a "minden termék" menüpontot. Itt lehetősége van végigtekinteni a jelenlegi árukészleten.

Amennyiben az "Aktualitások"-at választja, megtekintheti az éppen kiemelt ajánlatokat.

Ha a növényeket választja, megtekintheti a nem virágos készletet, ennek ellenkezője pedig a Virágok menüpont, ahol csak a virágzótermékek közül válogathat.

A Bejelentkezés/Regisztráció menüpontnál lehetősége van bejelentkezni vagy regisztrálni. Ha az előbbi választja, úgy már az elmentett adatokkal szabadon vásárolhat. A Profilom menüpont alatt, ha kiválasztja az Adatlapom menüpontot, akkor áttekintheti a regisztrációnál felvitt adatait vagy módosíthatja azokat. A kosár menüpontra kattintva áttekintheti a kosárba rakott termékeket, azok mennyiségét, igény szerint törölheti őket.

4.1 A program célja és lényegesebb funkciói

A program célja az, hogy legyen egy szemre könnyű, emészthető és könnyen kezelhető oldal. A lényegesebb funkciói a vásárlás, bejelentkezés, regisztráció, admin. Az admin fontos feladatokat lát el: adatokat visz fel (például: növény, termék), admin jogosultságot ad a felhasználóknak, törölhet felhasználót.

4.2 Futtatási környezet

Az alkalmazásunk webes környezetben futtatható. Használatához asztali számítógépet ajánlunk, főként erre a felületre terveztük, de a telefonos kinézete is rendesen használható.

4.3 A program fő funkcióinak leírása

4.3.1 A vásárló funkciói

REGISZTRÁCIÓ: regisztrálhatja magát a vendég, adatait (nevét, címét) a felületen rögzíti. Ezzel biztosítja a lehetőséget a vásárlásra. A regisztrációhoz a Laravel Breeze

API-t használtuk amely lehetővé teszi a felhasználók számára a regisztrációt, bejelentkeztetését és kezelését az alkalmazás API-ján keresztül

BEJELENTKEZÉS: szamon tartja a vásárlót és annak adatait, elérhetővé teszi a vásárlást. A Laravel Breeze API autentikáció segítségével a felhasználók be tudnak jelentkezni az alkalmazásba az API-ján keresztül, ami lehetővé teszi az engedélyezett hozzáférést a backend funkcionálisokhoz. Ez az adminok számára rendkívül fontos, hiszen ez dönti el mely feladatokat tudják elvégezni.

JELSZÓ MÓDOSÍTÁSA: Ez lehetővé teszi a bejelentkezett felhasználó számára, hogy az aktuális jelszó megadásával megváltoztathassa a jelszavát. Fontos megemlíteni, hogy addig nem engedi megváltoztatni a jelszót, amíg az új jelszó nem egyezik meg az ellenőrző jelszóval. Ez a funkció a felhasználó Adataim menüpontjában található.

KOSÁR: szamon tartja a vásárolni kívánt mennyiséget adott termékből. Továbbá előre kiszámolja a végösszeget is. A felhasználó írhatja a kosarat kívánsága szerint. Az egész kosarat lehet üríteni, de a tételeket egyenként is törölhetjük. (Session lejártá után ürül, nem tárolja a végtelenségig a vásárlást)

VÁSÁRLÁS: a megvásárolni kívánt termékeket kosárba rakhatjuk és megvásárolhatjuk igény szerint. Ezeknek a kosárba teendő mennyiségét változtathatjuk az aktuális termék oldalán. A vásárlást egy felugró ablakkal meg kell erősíteni, ha ez megtörtént, akkor a kosár automatikusan kiürül.

RÉGI RENDELÉSEK VISSZANÉZÉSE: Ezt a funkciót egy bejelentkezett felhasználó a Rendeléseim menüpont alatt találja meg. Itt minden különböző rendelését egy táblázatba tároljuk a kényelmes visszanezésért. Az itt megjelenő termékek neve vásárolt mennyisége és éppen aktuális ára van eltárolva, továbbá a végösszeget is vissza lehet nézni.

4.3.2 Az admin funkciói:

NÖVÉNY ADATOK FELVITELE: itt tud az admin új növényfajtákat felvinni tudományos/latin névvel és köznévvvel. Ezzel együtt bekerül az adatbázisba is.

KISZERELÉS FELVITELE: különböző kisereléseket adhatunk meg, szöveges formátumban röviden, lényegre törően.

FELHASZNÁLÓK MÓDOSÍTÁSA: az admin kibővítheti a felhasználó jogosultsági körét, azaz admin jogosultságot adhat neki.(törlés? – felhasználó adatainak módosítása?)

TERMÉKEK ÖSSZEÁLLÍTÁSA: a termékeket össze kell állítani a megfelelő adatokkal. A terméknek van: kiserelése, induló készlet mennyisége (ami beszerzéskor és eladásakor is változik)

TERMÉKEK MÓDOSÍTÁSA: a termékeket tudja módosítani az admin. Itt lehet szerkeszteni a termékeket. Itt lehet törölni a termékeket. (Ez kitörli a termékeket a termék táblából, jól jöhet, ha megszüntettünk egy terméket vagy rossz adatokkal vittük fel.)

ADMINJOGOSULTSÁG: adminjogosultságot adhat a regisztrált felhasználónak, például egy új kolléga esetén a kollégát regisztráljuk és az adminunk ad neki jogosultságot, így tudja használni a fontos funkciókat a cég részéről.

FELHASZNÁLÓK KEZELÉSE: a felhasználók kezelése elsősorban a törlésre vonatkozik, de a későbbiekben lehetőség szerint kérsre módosíthatja a felhasználó adatait.
(FONTOS! CSAK FELHASZNÁLÓI KÉRELEM ESETÉN MÓDOSTUNK ADATOT).

5 Fejlesztői dokumentáció

5.1 Fejlesztői környezet

Laravel 10 verzió

A Laravel egy nyílt forráskódú PHP alapú keretrendszer, amelyet webes alkalmazások készítésére használunk. Az általunk választott 10. verzió a legfrissebb funkciókat és fejlesztéseket kínálja, lehetővé téve a gyors és hatékony alkalmazásfejlesztést.

React 18.3.1 verzió

A React egy népszerű JavaScript könyvtár, amelyet felhasználói felületek készítésére használunk. Az általunk választott 18.3.1 verzió a legújabb funkciókat és fejlesztéseket kínálja, lehetővé téve a modern, dinamikus és skálázható felhasználói élmények létrehozását.

MySQL adatbázis

Az adatbázis-kezelő rendszerünk a MySQL, amely egy ingyenes és nyílt forráskódú relációs adatbázis-kezelő rendszer. A MySQL legfrissebb verzióját alkalmazzuk, hogy biztosítsuk az adatbázisrendszer stabilitását és biztonságát. Segítségével hatékonyan tárolhatunk és kezelhetünk nagy mennyiségű adatot a projektünk számára.

5.2 Program struktúra

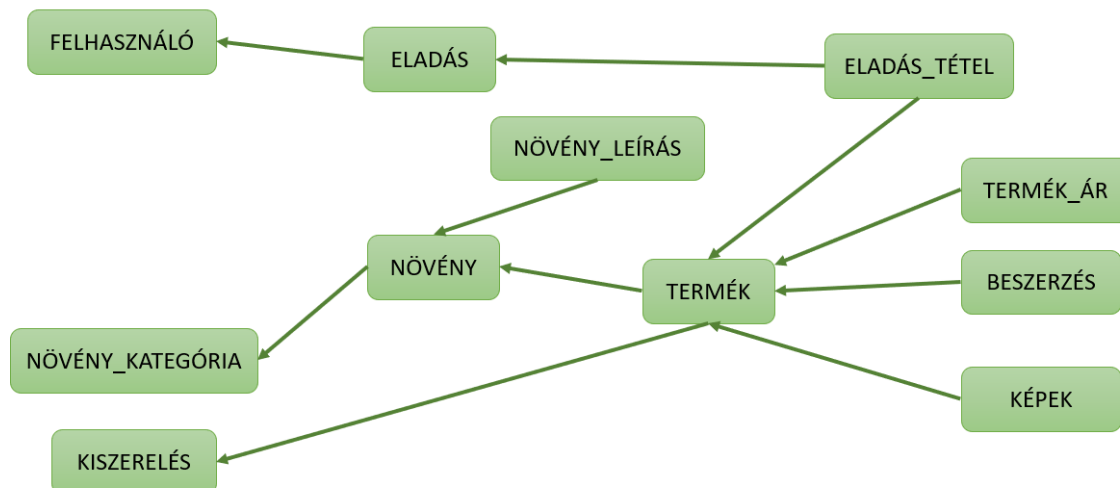
MVC

Az MVC egy architektúrális tervezési minta, melyet a program strukturált és jól szervezett felépítésére alkalmazunk. A Modell réteg az adatok kezeléséért és az üzleti logika végrehajtásáért felelős, a Nézet réteg pedig a felhasználói felület megjelenítését és az interakció lehetőségét biztosítja. A Vezérlő réteg pedig összeköti a Modellt és a Nézetet, kezelve a felhasználói interakciókat.

Az MVC struktúra segítségével könnyen megkülönböztethetjük az egyes részeket, és lehetőségünk van azokat elkülönítve fejleszteni és tesztelni. Ez átláthatóbbá teszi az alkalmazás kódját, hozzájárulva annak tisztaságához és könnyebb karbantarthatóságához. Emellett segít az alkalmazás kódjának hatékonyabb fejlesztésében és skálázhatóságában.

5.3 Adatbázis

5.3.1 Egyed-kapcsolati diagram



5.3.2 Az adatrendszer ismertetése

A tárgyalt rendszerben igyekszünk kényelmes és átlátható termékrendszert biztosítani mind a vásárlónak, mind az adminoknak.

- A *termék* nyilvántartja a növényt és annak tulajdonságait, árral és készlet mennyiséggel eltárolva. A *termék*ben számon tartjuk, hogy hány darabot foglaltak már le a készletből, így az aktuális készletből kivonva ellenőrizhetjük, hogy van-e elég készletünk a teljesítéshez.
- A *növény*ben megtalálhatóak a fajták latinnévvel azonosítva és növénykategóriával számon tartva.
- A *növénykategória*ban tároljuk a növény fajtáját.
- A *kiszerelés* tartja számon a növény méretét.
- A *termék ár* követi a termék árának változását dátummal, rögzíti adott dátumtól az új árat.
- A *felhasználó* tárolja a regisztrált felhasználóinkat, admint és vásárlót egyaránt, és a hozzáférésük szintjét.
- A *beszerzés* tábla kezeli a beszerzés dátumát darabszámmal és beszerzési árral.

- *Eladása tétel* tábla számon tartja az adott nyugtához tartozó terméket és a rendelt mennyiséget az adott tételből.
- Az *eladás* a kiszámolt végösszeg dátummal ellátva szerepel, a végösszeget az *eladástételen* szereplő termék kódhoz tartozó darab és a *termékben* szereplő azonos termékkódhoz tartozó ár alapján számoljuk ki.

5.3.3 Az ügyviteli funkciók

TERMÉKBESZERZÉS

- napibeszerzések rögzítése, a pillanatnyi készlet frissítésével

NÖVÉNYDEFINIÁLÁS

- növények számontartása
- növényfajta meghatározása

TERMÉKDEFINIÁLÁS

- termék karbantartás
- készlet és lefoglalt(kosárba helyezett) db nyomon követése

TERMÉKÁR

- termék árának nyomon követhető változtatása

NYUGTÁZÁS/ ELADÁS

- a kosár alapján létrejött vásárlás nyugtázása
- egy nyugtán szereplő összes tétel darabszám alapján kiszámolt árának összesítése

FELHASZNÁLÓK

- felhasználók karbantartása
- jogosultság megítélése

5.3.4 Az adatbázis-tervezés a szükséges egyedek kialakításával

(jelölés: TÁBLA (kulcs, leíró, *kapcsolathordozó*)

FELHASZNÁLÓ: a felhasználók eltárolása

FELHASZNÁLÓ(felhasználó_azon, név, email_cím, jelszó, felhasználó_sztint)

A felhasználók rendelkezhetnek jogosultsági szinttel, ezt a superadmin tudja csak kezelni. Bárkinek adhat admin jogosultságot. Adminok kezelik a beszerzést, a termékek felvitelét.

NÖVÉNY_KATEGÓRIA: növények hovatartozásának meghatározása elnevezéssel

NÖVÉNY_KATEGÓRIA(növény_kategória, elnevezés)

Ezek rövid elnevezések lesznek, mint pl.: Egynyáriak, Rózsák ,Bogyós gyümölcsök...

NÖVÉNY: latinnévvel, névvel és növény kategóriájának besorolásával számontartva

NÖVÉNY(tudományos_név, név, növény_kategória)

TERMÉK: az eladható termékek törzse az eladási árral és a már lefoglalt mennyiséggel

TERMÉK (termék_kód, növény, állapot, típus, szín, kiszereelés, ár, készlet, lefoglalt_mennyiség, szezonális)

Az állapot igen-nem típussal dönti el, hogy az adott növény milyen állapotban van, az igen az élőnövényt jelöli, ha nem akkor a magállapotot. Hasonlóan a típus is így dönti el, hogy dísz vagy haszon növényről van-e szó. Ha igen akkor dísznövény, ha nem akkor haszonnövény. Az alapállapot: mag és haszonnövény.

A készlet a termék éppen aktuális mennyiségét mutatja meg. Ha vásárlás történik, akkor elvesszük a mennyiséget, és ha beszerzés akkor automatikusan hozzáadódik az új árú.

A lefoglalt mennyiség akkor kerül használatba, ha egy vásárló kosárba tesz bizonyos mennyiséget egy adott termékből. Itt lefoglalja a megadott mennyiséget, hogy mindenképpen meg lehessen vásárolni, ez úgy történik hogy levonja az aktuális készletből a megadott mennyiséget.

A szezonális egy igen nem típusú mezővel dönti el, hogy az adott termék szezonális e.

KISZERELÉS: termékek kiszерelésének törzse, rövid megnevezéssel

KISZERELÉS(kiszerezés, név)

TERMÉK_ÁR: termékek tervezett árai sávosan tárolva

TERMÉK_ÁR(termék, mikortól, új_ár)

BESZERZÉS: termékek napi beszerzése

BESZERZÉS(termék_kód, beszerzés_dátuma, darabszám, beszerzési_ár)

ELADÁS: Az eladás törzsadatai, vásárlási dátummal és a kiszámolt fizetendő végösszeggel

ELADÁS(eladás_szám, vásárlás_dátuma, végösszeg)

ELADÁS_TÉTEL: az eladás_számmal tartozó tételek darabszámmal

ELADÁS_TÉTEL(eladás_szám, termék_kód, darabszám)

NÖVENY_LEÍRAS: a növényekhez tartozó leírásokat tartalmazza

NÖVENY_LEÍRAS (leírás_id, növény_kód, leírás)

KÉPEK: a termékekhez tartozó fényképek elérési útvonalát tárolja, a cél dönti el egy konkrét kép felhasználási célját.

A cél mező lehet például B ami borítóképre utal.

KÉPEK (termék_kód, elérési_útvonal,cél)

5.3.5 Táblák

K: kötelező

G: generált

Felhasználó

Oszlopnév	Típus	Szerep	Korlátozás
felhasználó_azón	int	elsődleges kulcs G	10000 <=
nev	varchar(30)	leíró, K	-
email	nvarchar(50)	leíró, K	-
jelszo	varchar(30)	K	-
felhaszn_szint	int	leíró, K	default(2), 0,1, 2

Novény Kategória

Oszlopnév	Típus	Szerep	Korlátozás
id	int	elsődleges kulcs G	10<=
elnevezes	nvarchar(30)	leíró, K	-

Novény

Oszlopnév	Típus	Szerep	Korlátozás
tudományos_nev	nvarchar(30)	elsődleges kulcs	-
nev	nvarchar(30)	leíró, K	-
novény_kategória	int	külső kulcs, K	-

Termek

Oszlopnév	Típus	Szerep	Korlátozás
termek_kod	int	elsődleges kulcs G	1000 <=
noveny	nvarchar(30)	külső kulcs	-
allapot	bit	leíró, K	--
tipus	bit	leíró, K	
szin	nvarchar(30)	leíró,	-
kiszereles	int	külső kulcs	-
ar	int	leíró, K	1 <= ar
keszlet	int	leíró, K	0 <= keszlet
lefoglalt_mennyiseg	int	leíró, K	0 <= lefoglalt_mennyiseg
szezonlis	bit	leíró, K	-

Kiszereles

Oszlopnév	Típus	Szerep	Korlátozás
id	int	elsődleges kulcs G	200 < =
nev	nvarchar(30)	leíró, K	-

Termek Ar

Oszlopnév	Típus	Szerep	Korlátozás
termek	int	külső kulcs	-
mikortol	date	leíró, K	-
uj_ar	int	leíró, K	1 <= uj_ar

Beszerzes

Oszlopnév	Típus	Szerep	Korlátozás
termek	int	külső kulcs	-
besz_datum	date	leíró, K	-
darabszam	int	leíró, K	1 <= darabszam default(1)
besz_ar	int	leíró, K	1 <= besz_ar

Eladas

Oszlopnév	Típus	Szerep	Korlátozás
eladas_szam	int	elsődleges kulcs	100000 <=
vasarlo	int	külső kulcs	
vas_datum	date	leíró, K	
vegosszeg	int	leíró, K	0 < vegosszeg

Eladas Tetel

Oszlopnév	Típus	Szerep	Korlátozás
eladas_szam	int	külső kulcs	-
termek	int	külső kulcs	-
darabszam	int	leíró, K	1 <= darabszam

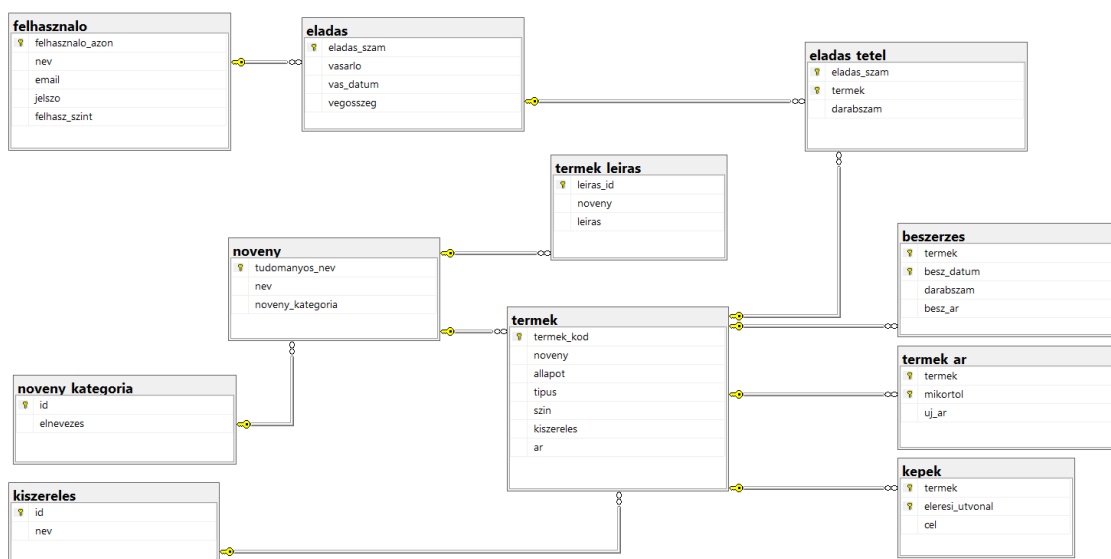
Noveny Leiras

Oszlopnév	Típus	Szerep	Korlátozás
leiras_id	int	elsődleges kulcs G	1 <=
noveny	nvarchar(30)	külső kulcs	-
leiras	nvarchar(MAX)	leíró, K	-

Kepek

Oszlopnév	Típus	Szerep	Korlátozás
termek	int	külső kulcs	-
eleresi_utvonal	nvarchar(60)	leíró, K	-
cel	nvarchar(2)	leíró, K	-

5.3.6 Kapcsolati ábra



5.3.7 Hasznos Lekérdezések

- Adott felhasználó mely termékeket vásárolta meg eddig. (Mikor mit mennyit mennyiért)
 - ezt a bejelentkezett felhasználó és az admin láthatja
(lehet nézet is, és akkor egyszerűbb megoldani, és majd le lehet rendezni frontenden)
- Egy adott termékből mely színek voltak a legkelendőek az elmúlt évben?
- Egy adott termékből mely kiszerelések voltak a legkelendőbbek az elmúlt évben?
- Adott termék mely hónapban volt a legkelendőbb az elmúlt évben
- A top 3 legkelendőbb termék az elmúlt hónapban.
- A legkelendőbb termék a mai nap folyamán.
- A hét legjobban fogyó terméke.
- Mely termékből nem fogyott az elmúlt hónapban.

9. Mely termékekből fogyott kevesebb mint az előző hónapban a legutolsó árváltozás óta?
10. Adott regisztrált felhasználónak mely növénykategóriák reklámozhatók az eddigi rendelései alapján?
 - a. adjon vissza konkrét terméket(ket).
11. Adott regisztrált felhasználó mely növénykategóriákból nem rendelt még soha?
12. Aktuális átlagár kiszámítása a beszerzési ár alapján, termékekre lebontva, tartalmazza a beszerzési árat, a jelenlegi eladási árat
13. A(z) ____ évi kiadások összege hónapra lebontva.
14. Az idei kiadások összege havi bontásban
15. Átlagosan hány százalékkal változtak meg a termékárak az előző évhez képest?
16. Mely termékekhez mely képek tartoznak?

5.3.8 Kiemelt hasznos lekérdezések

5. A top 3 legkelendőbb termék az elmúlt hónapban.

```
Select TOP 3 with ties et.termek, SUM(et.darabszam)
from eladas_tetel et
      inner join eladas e on e.eladas_szam=et.eladas_szam
where MONTH(e.vas_datum)=MONTH(GETDATE())-1
group by et.termek
order by 2 desc
```

8. Mely termékből nem fogyott az elmúlt hónapban.

```
go
create view multhonapi_eladasok
as
select termék, vas_datum
from eladas_tetel et
      inner join eladas e on e.eladas_szam=et.eladas_szam
where MONTH(e.vas_datum)=MONTH(GETDATE())-1
go
```

```

Select t.termek_kod
from termek t
      left outer join multhonapi_eladasok m on t.termek_kod=m.termek
where m.vas_datum is NULL

```

13. A(z) ____ évi kiadások összege hónapra lebontva.

```

go
CREATE function melyik_evi_kiadas
(
  @ev int
)
returns table
as
return
(
  select MONTH(besz_datum) as hónap, SUM(besz_ar * darabszam)
  from beszerzes
  where year(besz_datum)=@ev
  group by MONTH(besz_datum)
)
go

```

14. Az idei kiadások összege havi bontásban

```

Select MONTH(besz_datum) as hónap, SUM(besz_ar * darabszam)
from beszerzes
where year(besz_datum)=year(getdate())
group by MONTH(besz_datum)

```

5.3.9 Beépítendő megszorítások

1. Egyediség (kulcsok)
2. Hivatkozások (külső kulcsok)
3. Árak, készlet, új ár és a lefoglalt mennyiség nem negatív egész számok
4. Végösszegek a nyugtákon számított mezők
5. Egy termék aktuális árának megváltozásakor a régi ár elmentésre kerül

6. Ha egy termék bekerült a kosárba, akkor azt a termék táblába le kell vonni a készletből, és el kell tárolni a lefoglalt mennyiségben a megvásárlás pillanatáig.
7. Ha egy terméket megvásárolnak, akkor az le legyen vonva a készletből.
8. Kosárba tételkor ellenőrizni kell, hogy van-e a megvásárolni készült mennyiség raktáron.
9. Termék ne kerülhessen ki az oldalra készlet nélkül. (vagy jelezze hogy nincs készlet)
10. Termékek nem törölhetők, ha van termék a kosárban.
11. Egy előre meghatározott esemény (vásárlás gomb) történésére létrehozza az eladás számot az aktuális dátummal.

5.3.10 Technikai táblák (nem kötelezően megvalósítandó)

- kosár tábla és kosár tétel tábla, hogy az aktuális éppen zajló vásárlást számon tudja tartani a vásárló, de egy idő elteltével(session) a kosarat és a kosárban szereplő tételeket ürítjük.
- mentés, archiválás módja (a táblák sorainak szabályozott törlése)
- felkészülés a többnyelvűsége (az alapértelmezett nyelvű felh. funkcióknak és megjelenítendő adatoknak a más nyelvű megfelelői)

5.4 Backend

A backend elkészítéséhez Laravel keretrendszert használtunk, ami PHP alapú webes keretrendszer. Ez az MVC (Model-View-Controller) tervezési mintát használja, aminek célja a szerepkörök szétválasztása.

5.4.1 Modellek és Controllerek

5.4.1.1 Model

Feladata az adatok kezelése, adatstruktúrák meghatározása, létrehozása. Itt hajtjuk végre az adatokon elvégzendő műveleteket, mint a tárolásuk, és visszaolvasásuk. A model kommunikál az adatbázissal.

Saját Modeljeink

Model	Feladata/Leírás
User	A felhasználók adatait tárolja, a permission mező határozza meg az adott személy szerepkörét: Felhasználó, Admin, SuperAdmin.
PlantCategory	A növényekhez tartozó kategóriákat tárolja el, mint például: rózsafélék.
Plant	A növényeket tárolja, a latin nevük által vannak megkülönböztetve.
Unit	Az egyes termékek lehetséges kiszerezéseit tárolja el. Mint például: cserepes.
Product	A konkrét terméket tárolja el.
Purchase	A vásárlás adatait tárolja és kezeli, meghatározza mely Felhasználó vásárolt, és mikor.
PurchaseItem	Egy konkrét Terméket köt össze egy Vásárlással, tárolja az adott Termék darabszámát.
Suppliance	A Termékek beszerzését kezeli.
ProductPrice	A Termékek árváltozásait kezeli.
Description	A Termékekhez biztosít leírást.
Picture	A Termékekhez tartozó fényképek elérési útvonalát tárolja.

5.4.1.2 Controller

A Controller feladata a Model és a View közötti kapcsolat biztosítása. Egy API hívása során megadjuk, hogy mely Controller mely metódusa fusson le.

```
Route::get('konkret_product/{product_id}', [ProductController::class, 'konkretTermekKeppel']);
```

Saját Controllereink

- **PRODUCTCONTROLLER**

A ProductController egy olyan osztály, amely az alkalmazásban a termékekkel kapcsolatos műveletek végrehajtásáért felelős.

- index(): Ez a függvény visszaadja az összes terméket egy listában.
- show(\$id): Ezzel a függvénnyel egy adott termék adatait lehet lekérni az azonosítója alapján.
- update(Request \$request, \$id): A függvény frissíti a megadott azonosítójú termék adatait a kapott kérés alapján.
- store(Request \$request): Ez a függvény új terméket hoz létre a kapott adatok alapján és elmenti az adatbázisba.

- **USERCONTROLLER**

A UserController felelős az alkalmazás felhasználókkal kapcsolatos műveletek végrehajtásáért. A következő funkciók lehetővé teszik a felhasználók kezelését az alkalmazásban.

- index(): Visszaadja az összes felhasználót egy listában.
- show(\$id): Ezzel a függvénnyel egy adott felhasználó adatait lehet lekérni az azonosítója alapján.
- update(Request \$request, \$id): Frissíti a megadott azonosítójú felhasználó adatait a kapott kérés alapján.
- store(Request \$request): Új felhasználót hoz létre a kapott adatok alapján és elmenti az adatbázisba.

- **PLANTCONTROLLER**

A PlantController egy olyan osztály, amely az alkalmazásban a növényekkel kapcsolatos műveletek végrehajtásáért felelős.

- index(): Ez a függvény visszaadja az összes növényt egy listában. Az adminisztrációs felületen a növények listázásához használjuk.
- show(\$id): Ezzel a függvénnyel egy adott növény részletes adatait lehet lekérni az azonosítója alapján. Például egy specifikus növény adatait lehet megjeleníteni a felhasználói felületen.
- update(Request \$request, \$id): Frissíti a megadott azonosítójú növény adatait a kapott kérés alapján. Ez lehetővé teszi például a növény nevének, tudományos nevének vagy kategóriájának frissítését.

- `store(Request $request)`: Új növényt hoz létre a kapott adatok alapján és elmenti az adatbázisba. Egy űrlap kitöltésekor új növényt ad hozzá az adminfelületen.

• **PURCHASECONTROLLER**

A `PurchaseController` egy olyan osztály, amely az alkalmazásban a vásárlásokkal kapcsolatos műveletek végrehajtásáért felelős.

- `index()`: Visszaadja az összes vásárlást egy listában. Ez hasznos lehet például az adminisztrációs felületen a vásárlások listázásához.
- `show($id)`: Ezzel a függvénnyel egy adott vásárlás részletes adatait lehet lekérni az azonosítója alapján. Például a vásárlás részleteinek megjelenítésére a felhasználói felületen.
- `update(Request $request, $id)`: Frissíti a megadott azonosítójú vásárlás adatait a kapott kérés alapján. Ez lehetővé teszi például a vásárló nevének vagy a vásárlás dátumának frissítését.
- `store(Request $request)`: Új vásárlást hoz létre a kapott adatok alapján és elmenti az adatbázisba. Hasznos lehet például egy űrlap kitöltésekor az új vásárlás hozzáadásához.

Ezek a funkciók lehetővé teszik a vásárlások kezelését az alkalmazásban.

• **PURCHASEITEMCONTROLLER**

A `PurchaseItemController` felelős a vásárlási tételekkel kapcsolatos műveletek végrehajtásáért.

- `index()`: Visszaadja az összes vásárlási tételt egy listában. Ez lehetővé teszi a vásárlási tételek listázását.
- `show($purchase_number, $product_id)`: Ezzel a függvénnyel egy adott vásárlási tétel részletes adatait lehet lekérni a vásárlás száma és a termék azonosítója alapján.
- `destroy($purchase_number, $product_id)`: Törli a megadott vásárlási tételt az adatbázisból, és visszaállítja a kapcsolódó termék raktárkészletét.
- `update(Request $request, $purchase_number, $product_id)`: Frissíti a megadott vásárlási tétel adatait a kapott kérés alapján, és frissíti a kapcsolódó termék raktárkészletét.

- `store(Request $request)`: Új vásárlási tételt hoz létre a kapott adatok alapján és elmenti az adatbázisba. Ellenőrzi, hogy van-e elegendő készlet a termékből, és frissíti a készletet a megfelelő mennyiséggel.

- **SUPPLIANCECONTROLLER**

A SupplianceController az alkalmazásban a beszerzésekkel kapcsolatos műveletek végrehajtásáért felelős.

- `index()`: Visszaadja az összes beszerzést egy listában. Hasznos lehet például az adminisztrációs felületen a beszerzések listázásához.
- `show($product, $suppliance_date)`: Ezzel a függvénnyel egy adott beszerzés részletes adatait lehet lekérni a termék és a beszerzés dátumának alapján.
- `destroy($product, $suppliance_date)`: Törli a megadott termékhez és beszerzés dátumához tartozó beszerzést az adatbázisból.
- `update(Request $request, $product, $suppliance_date)`: Frissíti a megadott termékhez és beszerzés dátumához tartozó beszerzés adatait a kapott kérés alapján.
- `store(Request $request)`: Új beszerzést hoz létre a kapott adatok alapján és elmenti az adatbázisba.

- **PRODUCTPRICECONTROLLER**

A ProductPriceController felelős az alkalmazásban a termékek árváltozásainak kezeléséért.

- `index()`: Visszaadja az összes termékár-változást egy listában. Hasznos lehet például az adminisztrációs felületen a termékárak listázásához.
- `show($id)`: Ezzel a függvénnyel egy adott termékár-változás részletes adatait lehet lekérni az azonosítója alapján.
- `destroy($id)`: Törli a megadott termékár-változást az adatbázisból.
- `update(Request $request, $id)`: Frissíti a megadott termékár-változás adatait a kapott kérés alapján.
- `store(Request $request)`: Új termékár-változást hoz létre a kapott adatok alapján és elmenti az adatbázisba.

- **UNITCONTROLLER**

A UnitController felelős a növények kiszerezéseivel kapcsolatos műveletek kezeléséért az alkalmazásban.

- `index()`: Visszaadja az összes kiszerelést egy listában. Hasznos lehet például a kiszerelések listázásához az adminisztrációs felületen.
- `show($id)`: Ezzel a függvénnyel egy adott kiszerelés részletes adatait lehet lekérni az azonosítója alapján.
- `update(Request $request, $id)`: Frissíti a megadott kiszerelés adatait a kapott kérés alapján.
- `store(Request $request)`: Új kiszerelést hoz létre a kapott adatok alapján és elmenti az adatbázisba.

- **PLANTCATEGORYCONTROLLER**

A `PlantCategoryController` felelős az növénykategóriák kezeléséért az alkalmazásban. Nézzük meg részletesebben a funkciókat:

- `index()`: Visszaadja az összes növénykategóriát egy listában. Ez hasznos lehet például a kategóriák listázásához a felhasználói felületen.
- `show($id)`: Visszaadja az adott növénykategória részletes adatait az azonosítója alapján.
- `update(Request $request, $id)`: Frissíti az adott növénykategória adatait a kapott kérés alapján.
- `store(Request $request)`: Új növénykategóriát hoz létre a kapott adatok alapján, majd elmenti az adatbázisba.

Ezen funkciók lehetővé teszik a növénykategóriák kezelését az alkalmazásban, így könnyen csoportosíthatod és kezelheted az egyes növényeket a megfelelő kategóriákba.

Laravel Api Breeze Controllerei

- `RegisteredUserController`
- `AuthenticatedSessionController`
- `ConfirmablePasswordController`
- `PasswordController`

5.4.2 API végpontok

Az **Application Programming Interface** kifejezés röviden **API** egy olyan alkalmazásprogramozási interfész, vagy felület, amely lehetővé teszi más

szoftverek számára, hogy adatokat kérdezzenek le módosítsanak vagy töröljenek, illetve funkciókat hívjanak elő.

GET: Lekéri az erőforrás reprezentációját a megadott URI-n keresztül. A válaszüzenet törzse tartalmazza a kért erőforrás részleteit.

POST: Egy új erőforrást hoz létre a megadott URI-n. A kérésüzenet törzse tartalmazza az új erőforrás részleteit.

PATCH: Egy erőforrás részleges frissítését hajtja végre. A kérés törzse megadja az erőforrásra alkalmazni kívánt módosításokat.

DELETE: Eltávolítja az erőforrást a megadott URI-n

PUT: A megadott URI-n létrehoz egy új erőforrást, vagy cseréli a meglévőt. A kérésüzenet törzse meghatározza a létrehozni vagy frissíteni kívánt erőforrást.

Útvonalak a Product-hoz

Kérés típusa	URI	Az útvonalhoz tartozó függvény feladata
GET	api/product_w_pictures	Az összes Terméket lekéri a borítóképekkel. Rendesen kiírja a kiszerezést, a növény nevét, és leírást is tartalmaz.
GET	api/konkret_product/{product_id}	Egy konkrét Termék releváns adatait adja vissza képek nélkül.
GET	api/k_product_pictures/{product_id}	Egy konkrét Termék összes képét adja vissza.
GET	api/products	Az összes terméket adja vissza a Termék törzsadataival.
GET	api/products/{product_id}	Egy konkrét Terméket ad vissza a törzsadatokkal.
GET	api/boritokep	Az összes Termék borítóképként megjelölt fényképét adja vissza.

GET	api/description/{product_id}	Egy konkrét Termékhez tartozó leírást adja vissza.
POST	api/products	Létrehoz egy új Terméket.
PATCH	api/products/{product_id}	Egy konkrét létező Termék frissítését teszi lehetővé.

Útvonalak a User-hez

Kérés típusa	URI	Az útvonalhoz tartozó függvény feladata
GET	api/users	Az összes létező User-t vissza adja minden adatával együtt.
GET	api/users/{ \$user_id }	Egy konkrét User-t ad vissza minden adtával.

Útvonalak a Purchase-hez

Kérés típusa	URI	Az útvonalhoz tartozó függvény feladata
GET	api/purchases	Lekéri az Összes Vásárlást.
GET	api/ purchases /{ purchase_id }	Lekér egy specifikus Vásárlást.
GET	api/ aktualis_vasarlas/{user_id}	Visszaadja egy konkrét felhasználó legutobbi vásárlásához tartozó purchase_number-t.
POST	api/ purchases	Új vásárlást hoz létre.
PATCH	api/ purchases/{purchase_id}	Egy meglévő vásár módosítása.

Útvonalak a PurchaseItem-hez

Kérés típusa	URI	Az útvonalhoz tartozó függvény feladata
GET	api/purchase_items	Visszaadja az összes létező Vásárlási tételt. Tartalmazza

		mely terméket és mennyit vásároltak.
GET	api/purchase_items/{purchase_number}/{product_id}	Visszaad egy specifikus Vásárlási tételt
GET	api/rendelesek/{user_id}	Egy Adtott Felhasználó összes rendelését adja vissza.
DELETE	api/purchase_items/{purchase_number}/{product_id}	Egy létező Vásárlási tételt töröl.
POST	api/purchase_items	Létrehoz egy új Vásárlási tételt.
PATCH	api/purchase_items/{purchase_number}/{product_id}	Egy meglévő Vásárlási tételt módosít.

Útvonalak a Suppliance-hez

Kérés típusa	URI	Az útvonalhoz tartozó függvény feladata
GET	api/ supliances	Megjeleníti az összes Beszerzést.
GET	api/ supliances/{product}/{suppliance_date}	Megjelenít egy specifikus Beszerzést.
DELETE	api/ supliances/{product}/{suppliance_date}	Egy létező Beszerzést töröl.
POST	api/ supliances	Új Beszerzést hoz létre.
PATCH	api/ supliances{product}/{suppliance_date}	Egy létező Beszerzést módosít.

5.5 Frontend

5.6 Reszponzivitás

A weboldalunk elkészítéséhez főként Bootstrap-et használtunk, ahol nagyobb kontrollra volt szükségünk ott css-sel kiegészítettük, vagy felülírtuk. Bizonyos esetekben, például a rendeléseim menüpontban megtalálható táblázathoz Flexbox-ot használtunk az oldal szebb kinézete érdekében

Bootstrap:

A Bootstrap egy népszerű, nyílt forráskódú front-end keretrendszer. Célja, hogy segítsen a webfejlesztőknek gyorsan és egyszerűen létrehozni reszponzív, mobilbarát weboldalakat. A Bootstrap előre megírt CSS és JavaScript komponenseket tartalmaz, mint például gombok, űrlapok, navigációs menük és táblázatok, amelyeket könnyen integrálhatunk a saját projektjeinkbe. Ezáltal minimalizálja a fejlesztési időt és növeli az alkalmazások egységességét. A Bootstrap a CSS keretrendszer, amely könnyen testre szabható és bővíthető saját stílusok és komponensek hozzáadásával.

CSS (Cascading Style Sheets):

A CSS egy stíluslap nyelv, amelyet weboldalak és webes alkalmazások megjelenésének formázására használnak. A CSS segítségével definiálhatjuk az elemek stílusát, mint például a betűtípust, a színeket, a méreteket, a távolságokat és a pozíciókat. Az elnevezés ("Cascading") azt jelenti, hogy a stílusok hierarchikusan öröklődnek az elemek között, így lehetővé téve az egységes és könnyen karbantartható stílusok létrehozását. A CSS alapvető fontosságú minden webfejlesztő számára, mivel lehetővé teszi a weboldalak megjelenésének testreszabását és az egyedi dizájn létrehozását.

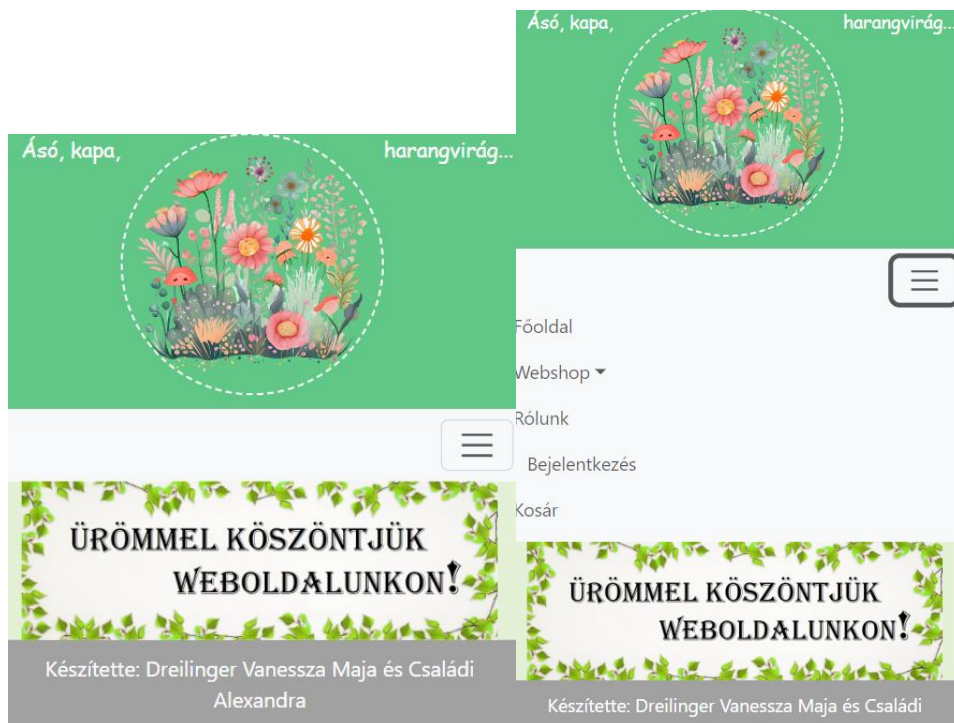
Flexbox:

A Flexbox (Flexible Box) egy CSS layout modell, amely lehetővé teszi a rugalmas és dinamikus elrendezéseket egy weboldalon vagy alkalmazásban. A Flexbox segítségével könnyen irányíthatjuk az elemek elhelyezkedését az egy dimenzió mentén, legyen az sor vagy oszlop. Ezáltal egyszerűbben kezelhetjük az elrendezést és a tartalom elosztását, optimalizálva a reszponzivitást és a különböző képernyőméretekhez való alkalmazkodást. A Flexbox különösen hasznos az elemek közötti egyenletes távolságok és rendezett elrendezések létrehozásában.

Főoldal



Mobil nézet



Bejelentkezés/ Regisztráció

Ásó, kapa, harangvirág...

Főoldal Webshop Rólunk Bejelentkezés Kosár

Bejelentkezés

Email:

Jelszó:

[Bejelentkezés](#)

Még nincs regisztrálva? [Regisztráció](#)

Készítette: Dreilinger Vanessza Maja és Családi Alexandra

Ásó, kapa, harangvirág...

Főoldal Webshop Rólunk Bejelentkezés Kosár

Regisztráció

Név:

Email:

Jelszó:

Jelszó újra:

[Regisztrálok](#)

Készítette: Dreilinger Vanessza Maja és Családi Alexandra

Mobil nézet

Ásó, kapa, harangvirág...

Regisztráció

Név:

Email:

Jelszó:

Jelszó újra:

[Regisztrálok](#)

Készítette: Dreilinger Vanessza Maja és Családi Alexandra

Bejelentkezés


Még nincs regisztrálva? [Regisztráció](#)

Készítette: Dreilinger Vanessza Maja és Családi Alexandra


Termékek oldal


[Főoldal](#) [Webshop](#) [Rólunk](#) [Bejelentkezés](#) [Kosár](#)

Részletes keresés
Search
Submit





Futómuskátli
Szín:
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)





kerti árvácska
Szín:
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)




Kerti bazsarózsa
Szín:
kiszerezés: 12 cm cserép
Ára: 690


 [Megnézem](#)




Munstead wood
Szín: vörös




Queen of Sweden
Szín: rózsaszín




Orvosi zsálya
Szín:





Rozmaring
Szín:
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)





Uborka
Szín:
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)





Cukkini
Szín: sárga
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)




Cukkini
Szín: zöld
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)




Cukkini
Szín: csikos
kiszerezés: 12 cm cserép
Ára: 690


 [Megnézem](#)

Készítette: Dreilinger Vanessza Maja és Családi Alexandra


Mobil nézet

Ásó, kapa, harangvirág...







Részletes keresés
Search
Submit



Futómuskátli
Szín:
kiszerezés: 12 cm cserép
Ára: 690



Kerti bazsarózsa
Szín:
kiszerezés: 12 cm cserép
Ára: 690

 [Megnézem](#)





Termék oldal

Főoldal Webshop Rólunk

Bejelentkezés Kosár

Kerti bazsarózsa

Paeonia officinalis



Kapható kiserelés: 12 cm cserép

Leírás

A Paeonia officinalis, vagy közismert nevén pünkösdi róza, elegancia és romantika szimbóluma. Csodálatos virágai, melyeket már évszázadok óta termesztnek, bármely kertet feldobják. Virágzása rövid, de az évek során egyre több hajtást hoz, így egyre impozánsabbá válik. Bámulatos illatával és lenyűgöző megjelenésével garantáltan lenyűgözi a látogatókat.

Elérhető készlet: 48 db

Ár: 690 Ft

-

1

+




Kosárba

Készítette: Dreilinger Vanessza Maja és Családi Alexandra



Mobil nézet

Kerti bazsarózsa

Paeonia officinalis



Kapható kiserelés: 12 cm cserép



Kapható kiserelés: 12 cm cserép

Leírás

A Paeonia officinalis, vagy közismert nevén pünkösdi róza, elegancia és romantika szimbóluma. Csodálatos virágai, melyeket már évszázadok óta termesztnek, bármely kertet feldobják. Virágzása rövid, de az évek során egyre több hajtást hoz, így egyre impozánsabbá válik. Bámulatos illatával és lenyűgöző megjelenésével garantáltan lenyűgözi a látogatókat.

Elérhető készlet: 48 db

Ár: 690 Ft

-

1




+

Kosárba

Készítette: Dreilinger Vanessza Maja és Családi Alexandra

Kosár

Kosár ürítése

	kerti árvácska	Mennyiség: 1 Ár: 690 Törölés
	Orvosi zsálya	Mennyiség: 2 Ár: 1380 Törölés
	Cukkini	Mennyiség: 7 Ár: 4830 Törölés

Végösszeg: 6900 Ft

Megvásárol

Mobil nézet

Cukkini






Mennyiség: 7 Ár: 4830 [Törölés](#)

Végösszeg: 6900 Ft

Megvásárol

Készítette: Dreilinger Vanessza Maja és Családi Alexandra

	kerti árvácska	Mennyiség: 1 Ár: 690 Törölés
	Orvosi zsálya	Mennyiség: 2 Ár: 1380 Törölés
	Cukkini	Mennyiség: 7 Ár: 4830 Törölés

Adataim

Szállítási információ

Név:

Cím:

Számlázási információ

Név:

Cím:

Frissítés

Jelszó változtatás

Jelenlegi jelszó:

Új jelszó:

Mobil nézet

Szállítási információ

Név:

Cím:

Számlázási információ

Név:

Cím:

Frissítés

Jelszó változtatás

Jelenlegi jelszó:

Rendeléseim

Főoldal

Webshop

Rólunk

Profilom

Kosár

Rendelés száma: 10000023

Termék neve	Mennyiség	Ár
Munstead wood	2	690
kerti árvácska	3	690
Futómuskátli	4	690
Összesen		6210

Rendelés száma: 10000025

Termék neve	Mennyiség	Ár
Uborka	6	690
Kerti bazsarózsa	1	690
Összesen		4830

Rendelés száma: 10000026

Mobil nézet

Rendelés száma: 10000023		
Termék neve	Mennyiség	Ár
Munstead wood	2	690
kerti árvácska	3	690
Futómuskátli	4	690
Összesen		6210
Rendelés száma: 10000025		
Termék neve	Mennyiség	Ár
Uborka	6	690
Kerti bazsarózsa	1	690
Összesen		4830
Rendelés száma: 10000026		
Termék neve	Mennyiség	Ár
Uborka	6	690

6 Tesztelés

TESZTELÉSI EGYSÉG

Bejelentkezés /Kijelentkezés és Regisztráció

AZ ÖSSZES TESZTESET

Teszteset / Bemeneti paraméterek	Várt eredmény	Kapott eredmény	Státusz
Bejelentkezés helyes adatokkal.	Sikeres bejelentkezés a megfelelő felhasználóval. A főoldalra irányít át.	A várt eredményt kaptuk. Sikeres bejelentkezés és átirányítás.	Elfogadva.
A kijelentkezés menüpontra kattintunk.	Sikeres kijelentkezés, local Storage-ből felhasználó eltávolítása.	Sikeres kijelentkezés, és a local Storage-t is üríti.	Elfogadva.
Bejelentkezés az adatbázisban nem létező email címmel.	Visszajelzést kapunk arról, hogy nem található ilyen felhasználó az adatbázisba. Bejelentkezés sikertelen lesz.	Nem enged bejelentkezni. Kapunk hibaüzenetet, de angolul. Az üzenet megegyezik a vártakkal.	Részlegesen elfogadva. Magyarosításra vár.
Bejelentkezés az hibás jelszóval.	Nem enged bejelentkezni. Hibaüzenetet ad vissza.	Nem lehet bejelentkezni, megfelelő hibaüzenetet kapunk vissza.	Részlegesen elfogadva. Magyarosításra vár.
Regisztráció megfelelő adatokkal.	Enged regisztrálni, új user-t hoz létre.	Sikeres regisztráció, új	Elfogadva

		felhasználó létrejön.	
Regisztráció már létező email címmel.	Nem enged regisztrálni, hibaüzenet arról, hogy az email-t már használják.	Nem lehet regisztrálni, megfelelő hibaüzenet.	Részlegesen elfogadva. Magyarosításra vár.
Regisztráció nem megegyező jelszavakkal.	Nem enged regisztrálni, hibaüzenetet kapunk erre vonatkozóan.	Sikertelen regisztráció, megfelelő hibaüzenetet kapunk.	Részlegesen elfogadva. Magyarosításra vár.
Regisztráció üresen hagyott mezővel.	Nem enged regisztrálni, hibaüzenetet kapunk erre vonatkozóan.	Nem lehet regisztrálni, jó hibaüzenetet kapunk.	Részlegesen elfogadva. Magyarosításra vár.

Termék kezelés felhasználóként

AZ ÖSSZES TESZTESZET

Teszteset / Bemeneti paraméterek	Várt eredmény	Kapott eredmény	Státusz
A Termékek oldalon rákattintunk a megnézem gombra. A termék megfelelően jelenik meg az oldalon.	Az adott terméknek megfelelő Konkrét Termék oldalra visz. A termék oldala megfelelően néz ki.	A megfelelő termék oldalára vitt. A megfelelő adatok jelennek meg a megfelelő módon.	Elfogadva
A Termék oldalán növelem, vagy csökkentem a kosárba teendő darab számot.	A darabszám Annak megfelelően növekszik vagy csökken, hogy	A vártaknak megfelelően működik a darabszám	Elfogadva

	hozzá akarok adni, vagy elvenni belőle.	mennyiségének változtatása.	
A kosárba gombra kattintunk miután kiválasztottuk a megvásárolni kívánt mennyiséget. Ezek után átnavigálunk a Kosár oldalára. Utólag még adunk az adott termékhez még X darabot.	A gombra kattintás után a Kosár oldalán megjelenik az előzőleg kiválasztott termék a megadott mennyiséggel. Az utólag hozzáadott mennyiség hozzáadódik a végső darabszámhoz.	A Kosár oldalán megjelent a megfelelő termék a megfelelő mennyiséggel. A darabszám a vártaknak megfelelően növekszik.	Elfogadva
A Kosár oldalon a végösszeg az összes kosárban lévő termék darabszám * árának összege.	A végösszeg megegyezik a kiszámoltakkal.	A végösszeg a várt összeg.	Elfogadva
Az egyik kiválasztott Kosárban lévő terméknel a Törlés gomb megnyomása.	Kitörli az adott terméket a Kosárból.	A vártaknak megfelelően törölte ki a Terméket.	Elfogadva
A kosár ürítése gomb megnyomása.	Az egész kosarat kiüríti.	A vártaknak megfelelően az egész kosarat kiürítette.	Elfogadva
A megvásárol gombra kattintás, és utána a mégsem gombra kattintunk.	Megjelenik egy felugró ablak, és a mégse gomb után egyszerűen visszatérünk a kosárba minden változás nélkül.	Megjelenik a felugró ablak, és aztán semmi nem történik.	Elfogadva

A megvásárol gombra kattintás, és utána a vásárlás véglegesítése gombra kattintunk.	Egy kis idő után a kosár teljesen kiürül. Ha az adatbázist megnézzük, akkor a releváns adatok bele kerültek. Vagy ha a Rendeléseim menüpontra megyünk, akkor ott megjeleni a leadott rendelés.	A kosár kiürül. Az adatbázis leellenőrizve bekerültek az adatok. A rendelések oldalon is minden megfelelően szerepel.	Elfogadva
---	--	---	-----------

Jelszó változtatás,

AZ ÖSSZES TESZTESZET

A jelenlegi jelszó megfelelő, az új jelszó, és a jelszó megerősítése megegyezik.	Sikeresen változik a jelszó	Sikeres jelszó változtatás.	Elfogadva
A jelenlegi jelszó megfelelő, az új jelszó, és a jelszó megerősítése nem egyezik meg.	Nem engedi a jelszó változtatását, hiba üzenetet ad vissza.	Nem lehet jelszót változtatni, a megfelelő hibaüzenet kaptuk.	Részlegesen elfogadva. Magyarosításra vár.
A jelenlegi jelszó nem megfelelő, az új jelszó, és a jelszó megerősítése megegyezik.	Nem engedi a jelszó változtatását, hiba üzenetet ad vissza.		Részlegesen elfogadva. Magyarosításra vár.

Admin

Egy termék adatainak módosítása az admin felületen	A kiválasztott termék módosított adata megváltozik.		Kijavításra vár.
Új növény hozzáadása a webshophoz. Az admin felületen.	a form-ba megadott adatok alapján az adatbázisba létrejön egy új növény		Kijavításra vár.
A felhasználók kilistázása csak superadmin részére.	Csak a superadminnak jelenjenek meg a felhasználók, sima admin ne férjen hozzá		Kijavításra vár.
Ha admin, vagy superadmin van bejelentkezve, csak akkor elérhető az admin felület.	Ha nincs megfelelő jogosultság, akkor az admin oldalt nem lehet elérni.	Amikor megpróbáljuk elérni, akkora nemtalálható oldalra irányít.	Elfogadva
Sima adminként a felhasználók kilistázásának megnézése.	Nincs jogosultság hozzá, nem lehet elérni.	Amikor megpróbáljuk elérni, akkora nemtalálható oldalra irányít.	Elfogadva

7 Fejlesztési lehetőségek

A következő fejlesztési lehetőségekre gondoltunk.

- A következőkben lehetne fájl beküldéssel feltölteni az egyes táblákat, például a növényeket vagy akár a kiszerezést.
- Kedvencek tábla és menüpont a felhasználónak. A felhasználó eltárolhatná kedvenc vagy megvásárolni kívánt termékeit későbbi vásárláshoz.
- Email küldés, hogy értesíthessük a felhasználókat.
- Készlet bővítés: egyéb termékek eladása növényeken kívül.
- Különleges eladás: az admin létrehozna egy eladást, ezek különleges eladások lennének: például saját részre eladni vagy áruházak közötti eladás zajlana.
- Keresések részletezése, hogy a felhasználó a lehető legpontosabban kereshessen egy termékre.
- Alkalmazottak kezelése: az alkalmazottakat külön kezelnék az egyszerű felhasználóktól. Ezt csak az admin vihetné fel és nem kellene a kommunális oldalon regisztrálni a kollégának.
- Selejtezés: hibás vagy kifutott termékek elkülönítése, ezeket tudnánk kedvezményesen kínálni és eladni
- a TERMÉK-et mozgathatjuk a raktárból a bemutató térbe, vagy leselejtezzük
- Hűség jutalmazása a felhasználónak: Pont gyűjtés/ törzsvásárlói kártya, pontok felhasználhatók ahhoz, hogy a végösszezből X összeg lejönnne.

8 Irodalomjegyzék

Hivatalos Dokumentációk:

<https://react.dev/learn>

<https://laravel.com/docs/10.x>

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>

Egyéb

<https://kiszervezettmarketing.hu/weboldal-keszites/api-jelentese/>

https://ade.web.elte.hu/wabp/lecke7_lap1.html