



Vas Megyei Szakképzési Centrum
Nádasy Tamás Technikum és Kollégium

SZAKDOLGOZAT

Shopper

**Bogács László Gyula, Ragasits Márk, Takács
Kevin Gábor**

**Konzulens:
Balics Gábor**

2023

Nyilatkozat

Alulírott, Bogács László Gyula, Ragasits Márk, Takács Kevin Gábor Szoftverfejlesztő és -tesztelő szakos hallgatók kijelentjük, hogy a Shopper című szakdolgozat feladat kidolgozása a saját munkánk, abban csak a megjelölt forrásokat, és a megjelölt mértékben használtuk fel, az idézés szabályainak megfelelően, a hivatkozások pontos megjelölésével.

Eredményeim saját munkán, számításokon, kutatáson, valós méréseken alapulnak, és a legjobb tudásom szerint hitelesek.

Csepreg, 2023.03.23

hallgató

Csepreg, 2023.03.23

hallgató

Csepreg, 2023.03.23

hallgató

Kivonat

Shopper

Az online vásárlás egyre népszerűbbé válik a digitális korban, ennek egyik fő oka az online vásárlási weboldalak és alkalmazások elterjedése. Ezek a platformok lehetővé teszik számunkra a kényelmes és hatékony vásárlást, és különböző funkciókkal és szolgáltatásokkal javítják vásárlási élményünket. Az online vásárlás azonban eszközöket is igényel a tervezéshez és a végrehajtáshoz. A Shopper megoldás erre a problémára, mivel az emberek vásárolhatnak anélkül, hogy elfelejtenének valamit, mert egyszerűen ellenőrizhetik a telefonjukat, hogy van-e hiányzó elem vagy sem. Ezenkívül a papíralapú vásárlói listák problémája az, hogy könnyen elveszhetnek vagy eltéphetnek, és kézzel kell írni őket, ami nehéz olvasni, ha valakinek a kézírása nem egyértelmű. A Shopper megoldja ezt a problémát, mivel ez egy weboldal a telefonodon, így valószínűtlen, hogy megsérül vagy elveszne a zsebedben, és könnyen olvasható.

Abstract

Shopper

Online shopping is becoming increasingly popular in the digital age, and one of the main reasons for this is the prevalence of online shopping websites and apps. These platforms allow us to shop conveniently and efficiently, and enhance our shopping experience with various features and services. However, online shopping also requires tools to help with planning and execution. Shopper is the solution to this problem as people can shop without forgetting something because they can just check their phone and see if anything is missing or not. Also, the problem with paper shopping lists is that they can easily get lost or torn, and they have to be written by hand, which can be difficult to read if someone's handwriting is not clear. Shopper solves this problem, as it is a website on your phone, so it is unlikely to get damaged or lost in your pocket, and it is easy to read.

Tartalomjegyzék

Abstract	4
1. Bevezetés	7
1.1. Bevásárló Listák online	7
2. A Shopinger weboldal bemutatása	9
2.1.Használt programozási nyelvek	9
2.1.1. HTML.....	9
2.1.2. CSS.....	10
2.1.3. Svelte.....	10
2.1.4. Typescript.....	11
2.1.5 Javascript	12
2.1.6 Prisma.....	12
2.2.1. Regisztráció.....	14
2.2.2. Bejelentkezés.....	14
2.2.3. Otthon.....	15
2.2.4. Menü	16
2.2.5. Lista hozzáadás.....	17
2.2.6. Lista	18
2.2.7. Felhasználó listához adás.....	18
2.2.8 Lista módosítás	19
2.2.9. Lista törlés.....	19
2.3 Adatbázis	20
3. Tesztelés.....	22
3.1 Bejelentkezés teszt	22
3.2 Regisztráció teszt.....	23

3.3 Lista teszt	24
3.3.1 Lista létrehozás	24
3.3.2 Megosztás	24
3.3.3 Módosítás	25
3.3.4 Lista törlése	26
3.4 Menü.....	27
Összegzés	28
Ábrajegyzék	29
Irodalmijegyzék.....	30
Mellékletek.....	31

1. Bevezetés

Az online vásárlás egyre népszerűbbé válik a digitális korban, és ennek egyik fő oka az online bevásárló weboldalak és alkalmazások elterjedése. Az ilyen oldalak és alkalmazások lehetővé teszik, hogy kényelmesen és hatékonyan vásároljunk, a vásárlási élményünket pedig számos funkcióval és szolgáltatással gazdagítják.

Azonban az online bevásárláshoz szükség van olyan eszközökre is, amelyek segítenek a bevásárlás megtervezésében és lebonyolításában. A Shopper erre a problémára ad megoldást, és lehetővé teszi, hogy az emberek kényelmesen és hatékonyan tervezhessék meg a bevásárlást, és a lehető legtöbb előnyt kínálják fel számukra.

A szakdolgozat részletesen bemutatja a Shopper tervezési folyamatát, beleértve a funkciók kidolgozását és a felhasználói felület megtervezését. A Shopper számos előnnyel jár, beleértve az idő megtakarítását, és a bevásárlólista könnyű kezelhetőségét.

A szakdolgozat célja, hogy bemutassa, hogyan lehet létrehozni egy hatékony és felhasználóbarát online bevásárlólista weboldalt, mint például a Shopper, amely hozzájárul a felhasználók életének egyszerűsítéséhez.

1.1. Bevásárló Listák online

Az online bevásárló listák egyre népszerűbbé válnak a digitális korban, és egyre több ember választja őket a hagyományos papír alapú listák helyett. Az online bevásárló listák számos előnyt kínálnak a hagyományos papír alapú listákhoz képest.

Az egyik fő előnye az online bevásárló listáknak az, hogy sokkal könnyebb kezelni őket, és lehetőséget adnak az egyszerű és hatékony bevásárlási tervezésre. Az online listák lehetővé teszik, hogy a felhasználók gyorsan és könnyen hozzá adhassanak vagy eltávolítsanak elemeket a listáról, és lehetővé teszik a bevásárló lista megosztását másokkal is, így könnyen koordinálható az egész család bevásárlása.

Az online bevásárló listák előnyei közé tartozik a környezetvédelem is, mivel csökkentik a papíralapú bevásárló listák használatát, és ezáltal csökkentik az általuk generált hulladékot.

Az online bevásárló listák azonban nem mindenki számára ideálisak. Az idősebb generáció tagjai például jobban szeretik a hagyományos papír alapú listákat, mivel számukra ezek szokásosak és ismerősek. Emellett az online listákhoz szükséges internetkapcsolat is korlátozhatja azok használatát.

Összességében az online bevásárló listák sok előnnyel járnak, és számos ember számára hatékony és praktikus megoldást kínálnak a bevásárlási tervezésre. Azonban fontos megemlíteni, hogy a hagyományos papír alapú listák is továbbra is népszerűek és hasznosak lehetnek, különösen azok számára, akik nem rendelkeznek internetkapcsolattal vagy azok számára, akik számára a hagyományos módszerek a megszokottak.

2. A Shopinger weboldal bemutatása

A Shopinger egy online bevásárló lista alkalmazás, amely lehetővé teszi a felhasználók számára, hogy egyszerűen és hatékonyan készítsenek bevásárló listákat, és ezeket könnyen megosszák barátaikkal és családjukkal. Az alkalmazás lehetővé teszi a felhasználók számára, hogy egyszerűen hozzáadjanak és eltávolítsanak termékeket a listáról, és csoportosítsák őket kategóriákba.

A Shopinger weboldal felhasználóbarát felületet kínál, amely lehetővé teszi a felhasználók számára, hogy könnyen regisztráljanak és bejelentkezzenek az alkalmazásba. Az oldal tartalmaz egy áttekintő panelt, amely megjeleníti a felhasználók összes bevásárló listáját, valamint lehetőséget biztosít az új lista létrehozására és a meglévő listák szerkesztésére.

Az alkalmazás és a weboldal biztonságos és megbízható, mivel az adatok titkosított kapcsolaton keresztül történő átvitele és tárolása megfelelően megvalósított.

A Shopinger weboldal bemutatása során részletesen leírjuk az alkalmazás és a weboldal funkcióit, azok használatát és a felhasználói élményt, amelyet az alkalmazás nyújt.

2.1. Használt programozási nyelvek

2.1.1. HTML

Az HTML (HyperText Markup Language) egy strukturált nyelv a weboldalak tartalmának leírására. A nyelv alapja a tag, amely az adott elem jellemzőit határozza meg. Az elemek közötti szöveg a tag-ek között helyezkedik el. Az HTML tartalmazza a szöveg formázásához használt stílusokat, képek, videók és hangfájlok beillesztését, valamint a linkek a weboldalak közötti navigációhoz. Az HTML-t a webböngészők értelmezik és megjelenítik. A webfejlesztéshez használt egyéb nyelvek és keretrendszerek kiegészítik az HTML-t és további lehetőségeket kínálnak a weboldalak tervezéséhez és fejlesztéséhez.



1. ábra HTML logo

2.1.2. CSS

A Cascading Style Sheets (CSS) egy olyan stíluslap nyelv, amely lehetővé teszi, hogy formázd az HTML-dokumentumokat. A CSS segítségével beállíthatod az elemek színét, betűméretét, elrendezését, és még sok minden mást. A CSS segít abban, hogy az oldalad jól strukturált legyen és könnyen olvasható legyen.

A CSS két fő részből áll: a szelektorból és a deklarációból. A szelektor az a HTML elem, amelyre a stílust alkalmazni szeretnéd, míg a deklaráció az a tulajdonság és az érték, amelyet be akarsz állítani.

A CSS-t használva lehetőség van arra, hogy elkülönítsd a tartalmat és a kinézetet. Ez azt jelenti, hogy az oldal tartalmát külön tudod választani a kinézettől, így könnyebbé válik az oldal karbantartása és átláthatóbbá válik a kód.

A CSS sok előnnyel rendelkezik, például egységes megjelenést biztosít az oldaladnak minden böngészőben, több oldalas oldalak kinézetének egységesítése, gyorsabb oldalbetöltés, könnyebb karbantarthatóság. Használatával lehetőség van arra, hogy az oldaladat testre szabd az adott célcsoportnak megfelelően és emellett növelheted az oldal felhasználói élményét.



2. ábra CSS logo

2.1.3. Svelte

A Svelte egy modern, nyílt forráskódú, kompakt keretrendszer, amelyet a webfejlesztők használhatnak webalkalmazások és komponensek készítéséhez. A Svelte arra törekszik, hogy könnyen tanulható és használható legyen, míg ugyanakkor nagyon hatékony és gyors marad. A Svelte egy reaktív keretrendszer, ami azt jelenti, hogy automatikusan frissíti az adatokat az alkalmazásban, ha azok változnak, így a felhasználói élmény sokkal simább és interaktívabb lehet.



3. ábra Svelte logo

A Svelte előnyei közé tartozik a szintaktikai egyszerűség, az alacsony méret, és az átlátható komponens alapú struktúra. Az alkalmazások kódjának mérete kisebb lehet a Svelte használatával, és az alkalmazások gyorsabbak lehetnek, mivel a Svelte a futásidőben optimalizálja a kódot. A Svelte emellett lehetővé teszi a webalkalmazások és komponensek könnyű testre szabását is.

A Svelte használata egyre népszerűbb a webfejlesztők körében, és számos nagyvállalat is alkalmazza a webalkalmazásaik és komponenseik készítése során. Ha a webalkalmazások gyors és hatékony fejlesztésére van szükség, a Svelte egy jó választás lehet.

2.1.4. Typescript

A TypeScript egy nyílt forráskódú programozási nyelv, amely az erősen típusos JavaScript nyelven alapul. A TypeScript célja, hogy nagyobb skálájú JavaScript alkalmazásokat készítsünk, amelyek könnyebben karbantarthatók és fejleszthetők.

A TypeScript a JavaScript bővítése, amely lehetővé teszi az erősebb típusellenőrzést és a fejlesztési időbeni hibák számos megelőzését. Az erős típusrendszer biztosítja,

hogy a kódunk konzisztens maradjon és biztonságosabb legyen a futásidőben.

A TypeScript fejlesztése a Microsoft által kezdődött 2012-ben, és azóta jelentős népszerűsége tett szert a fejlesztők körében. A TypeScript használható minden olyan projektben, amelyben a JavaScript is használható, és sok fejlesztő választja ezt a nyelvet, mert hatékonyabbá és egyszerűbbé teszi a fejlesztést.

A TypeScript egy újabb réteget ad a JavaScript-hez, amelynek segítségével könnyebben és biztonságosabban lehet fejleszteni. Használatával lehetőségünk van a kódunk hibáinak gyorsabb felfedezésére és javítására, ami időt és pénzt takaríthat meg számunkra.

A TypeScript használatával lehetőségünk van a moduláris kódolásra is, amely hatékonyabbá teszi a fejlesztést. Emellett lehetőségünk van a szerver-oldali programozásra is, ami az alkalmazásunk teljesítményét javíthatja.

Összességében a TypeScript egy hatékony és népszerű programozási nyelv, amelyet érdemes megfontolni, ha nagyobb skálájú JavaScript alkalmazásokat szeretnénk fejleszteni.



4. ábra Typescript

2.1.5 Javascript

Javascript egy nagyon népszerű programozási nyelv, amelyet elsősorban weboldalak fejlesztésére használnak. A nyelv lehetővé teszi a dinamikus weboldalak létrehozását és a felhasználói interakciók javítását, mint például animációk és űrlapok kezelése.

A Javascript futtatásához nincs szükség különösebb beállításokra, mivel minden modern böngésző támogatja. Ez lehetővé teszi, hogy a nyelvet könnyen és gyorsan használják a weboldalak fejlesztéséhez.

A Javascript egy nagyon rugalmas nyelv, amely lehetővé teszi a fejlesztők számára a kód bázis egyszerű bővítését és átalakítását. Ez nagyon fontos, mivel a weboldalak és az alkalmazások folyamatosan változnak és fejlődnek, és a fejlesztőknek alkalmazkodniuk kell a változásokhoz.

A Javascript nyelv folyamatosan fejlődik és új funkciókat és lehetőségeket adnak hozzá. Ennek köszönhetően a fejlesztőknek mindig van lehetőségük a kódjuk fejlesztésére és a lehetőségeik bővítésére.

Összességében, a Javascript egy nagyon hasznos és rugalmas programozási nyelv, amely lehetővé teszi a weboldalak és az alkalmazások kiváló minőségű fejlesztését és könnyen bővíthető.



5. ábra JavaScript logo

2.1.6 Prisma

Prisma egy adatbázis-leképező (ORM) eszköz, amely lehetővé teszi a fejlesztők számára, hogy egyszerűen és hatékonyan dolgozzanak az adatokkal a backend részben. Az ORM eszközök célja, hogy lehetővé tegyék az adatbázis-séma objektumorientált módon való kezelését, így lehetővé téve az alkalmazásoknak, hogy hatékonyan kommunikáljanak az adatbázisokkal.

A Prisma számos szolgáltatást kínál, például automatikus sémaleképezést és migrációt, képes adatbázis-séma

változásokat követni, és a Prisma Client által nyújtott nyelvi támogatás lehetővé teszi a



6. ábra Prisma logo

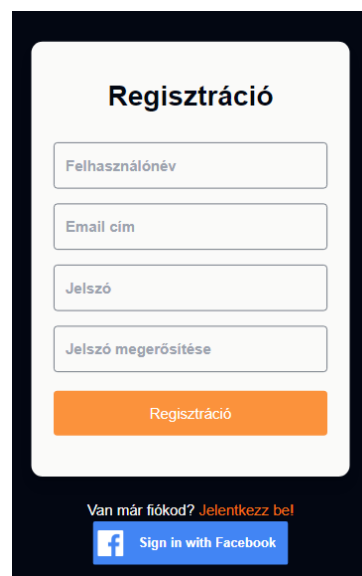
TypeScript használatát az alkalmazások fejlesztése során. A Prisma támogatja az adatbázisok közül a legnépszerűbbeket, például PostgreSQL, MySQL és SQLite.

A Prisma egy új generációs adatbázis-leképező eszköz, amely segítségével a fejlesztők hatékonyan tudják kezelni az adatbázisokat a backend oldalon. A Prisma rugalmas és könnyen használható, és számos előnyös funkcióval rendelkezik, például a szervezett adatkezelést és az automatizált sémaleképezést. A fejlesztők számára lehetővé teszi az objektumorientált adatkezelést, és a Prisma Client támogatja a TypeScript nyelvet is. Így a fejlesztők biztosak lehetnek abban, hogy hatékonyan dolgoznak az adatokkal a backend részben.

2.2. A felhasználói felület működése

2.2.1. Regisztráció

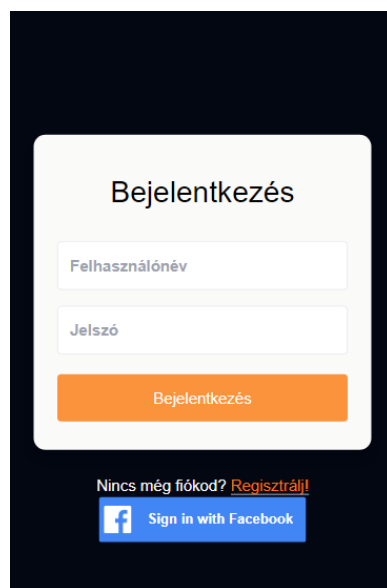
A regisztráció fontos eleme az online bevásárló listának, mivel itt hozzák létre a felhasználók a saját fiókjukat, amelybe később, a bejelentkezés után beléphetnek. A regisztráció során a felhasználóknak meg kell adniuk egy felhasználónevet, egy e-mail címet és egy jelszót, vagy akár Facebook fiókkal is be lehet jelentkezni. A jelszó legalább 8 karakter hosszú kell legyen. A regisztráció végén a felhasználók képesek lesznek létrehozni a saját bevásárló listájukat és használni az összes funkciót, amelyet a rendszer nyújt.



7. ábra Regisztrációs felület

2.2.2. Bejelentkezés

Az online bevásárló listán való bejelentkezés lehetővé teszi a felhasználók számára, hogy hozzáférjenek a saját fiókjához és az összes korábban létrehozott bevásárlólistához. A bejelentkezéshez szükséges az e-mail cím és a jelszó megadása. A felhasználók a Facebook fiókjukkal is bejelentkezhetnek, ha azt választják. A bejelentkezés után a felhasználók teljes körű hozzáférést kapnak a rendszer funkcióihoz, beleértve a listák kezelését, a bevásárlólisták megosztását más felhasználókkal és a bevásárlólisták automatikus elkészítését az előre megadott paraméterek alapján.



8. ábra Bejelentkezési felület

2.2.3. Otthon



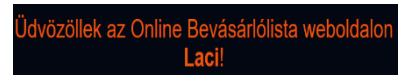
9. ábra Üdvözlő

Az otthoni felület az első benyomást kelti a felhasználóban a weboldalról, ezért rendkívül fontos, hogy megfelelően legyen kialakítva. Az otthoni felületen a felhasználóknak lehetősége van regisztrálni

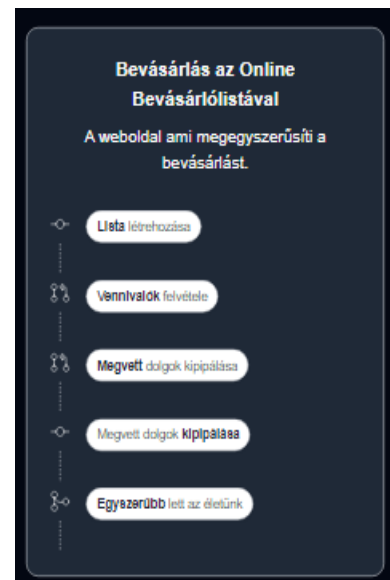
és bejelentkezni, amelyek kulcsfontosságúak a weboldal használatának megkezdése előtt. Az otthoni felületen található egy hasznos bemutató is, amelynek segítségével az új felhasználók könnyen megtanulhatják a weboldal használatát. A bemutató rövid és tömör, lépésről-lépésre bemutatva a weboldal működését.

Ezen kívül az otthoni felületen található fülek között szerepel a lista létrehozása, amely a weboldal egyik legfontosabb funkciója. Az otthoni felületen azonban nem csak ez az egy funkció található meg, hanem a menü is amin szerepel összes további funkció is, amelyek segítenek a hatékony és kényelmes weboldal használatban.

Az elégedett felhasználók alapja a felhasználóbarát felület és a könnyen érthető navigáció. Az otthoni felület tervezése és kialakítása lehetővé teszi, hogy az új felhasználók gyorsan megtanulják a weboldal használatát, míg a regisztrált felhasználók minden szükséges funkcióhoz egyszerűen és kényelmesen férhetnek hozzá az otthoni felületen keresztül. A felhasználóbarát felület és a könnyen érthető navigáció pedig elősegíti az elégedett felhasználók visszatérését és több vásárlást generál a weboldalon.



10. ábra Személyes üdvözlés



11. ábra Bemutató

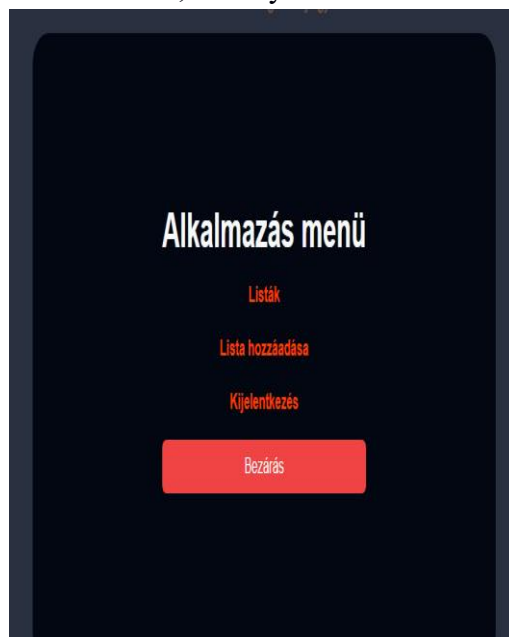
2.2.4. Menü



12. ábra Menü ikon

Az alkalmazás Menü ikonja szintén fontos elem az oldalon, amelyre kattintva megjelenik az alkalmazás menüje. Ezen menü segítségével a felhasználók számos lehetőség közül választhatnak, például elérhetik a Listák fülét, amelyen keresztül megtekinthetik, szerkeszthetik és törölhetik a meglévő listáikat, illetve létrehozhatnak újat. Emellett az alkalmazás menüjében megtalálható a Lista hozzáadása lehetőség, amelyen keresztül a felhasználók új listákat hozhatnak létre. Az alkalmazás menüjében elérhető a kijelentkezés opció is, amely segítségével a felhasználók biztonságosan kiléphetnek a fiókjukból, amikor az már nincs szükségük rá. Az alkalmazás menüjében található bezárás gomb pedig lehetővé teszi, hogy a felhasználók egyszerűen bezárják az alkalmazás menüjét, ha már megtalálták, amit kerestek.

Ha a felhasználó valamilyen okból másik részre szeretne navigálni az alkalmazásban, akkor egyszerűen rá kell kattintania a megfelelő szövegre, és az oldal automatikusan elnavigál a kiválasztott részre. Az alkalmazás menüje tehát lehetőséget kínál a felhasználóknak a fontos funkciók elérésére és az egyszerű navigációra az alkalmazásban.



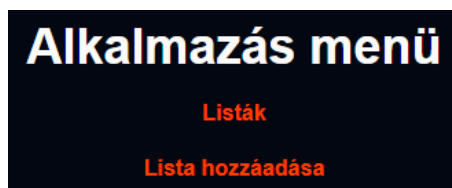
13. ábra Menü



14. ábra Otthoni listák

2.2.5. Lista hozzáadás

Az új lista létrehozásához a felhasználó a menüből elérhető "Lista hozzáadás" opciót használja. Az egyszerű és felhasználóbarát felületnek köszönhetően a lista elkészítése rendkívül egyszerű, és minden fontos információ gyorsan beírható. A lista neve kiválasztása után a felhasználó azonnal hozzáadhat elemeket a listához. Az elemek hozzáadása rendkívül egyszerű, csak be kell írni a nevüket vagy az összetevőket, mennyiséget vagy bármilyen egyéb fontos információt, majd kattintani a "Hozzáadás" gombra. Az elemek törlése szintén egyszerű: a felhasználónak csak a lista mellett található kockára kell kattintania az elem neve mellett. A lista elkészítése után a felhasználó bármikor szerkesztheti, módosíthatja vagy törölheti a lista tartalmát. Ez azt jelenti, hogy ha az élelmiszerboltban vásárlás közben eszébe jut valami, amit hozzá szeretne adni a listához, vagy ha egy elemet később már nem kell, egyszerűen szerkesztheti a listát az otthoni felületen keresztül. Az egyszerű, könnyen kezelhető és felhasználóbarát listakészítő funkció segítségével a felhasználók bármikor létrehozhatnak új listákat, és azokat gyorsan és hatékonyan szerkeszthetik és frissíthetik a megfelelő funkciók segítségével.



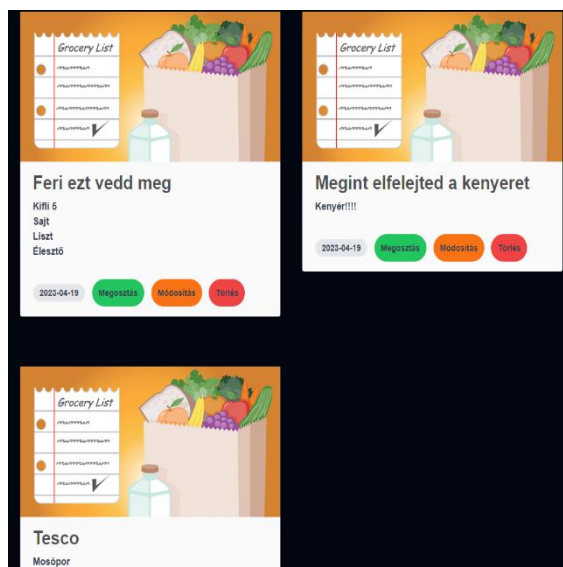
15. ábra Lista

A light-themed creation panel titled 'Létrehozás'. It contains a text input field with the placeholder 'Most nem felejteted el', another text input field with the placeholder 'Új vennivaló..', a green 'Add' button, a list of items with checkboxes, a date input field showing '2023-04-19', and an orange 'Hozzáadás' button. The list of items includes: '1 Kenyér', '1 Sajt', 'Macko sajt', '2 Tej', '5 Kifli', 'parizer', 'Viz', and 'Ami neked kell'.

16. ábra Lista létrehozási panel

2.2.6. Lista

A Listák funkció lehetővé teszi a felhasználók számára, hogy könnyen és hatékonyan készítsenek listákat a különböző feladatokról, például bevásárló listáról, teendőkről vagy bármilyen másról. A funkció elérhető a menüből, vagy gyorsan hozzáférhető a fejlécben található link segítségével. Az egyszerű és felhasználóbarát felületnek köszönhetően a lista létrehozása rendkívül egyszerű, és a felhasználók szabadon hozzáadhatják, szerkeszthetik vagy törölhetik a lista tartalmát. Emellett lehetőség van a lista megosztására más felhasználókkal. A

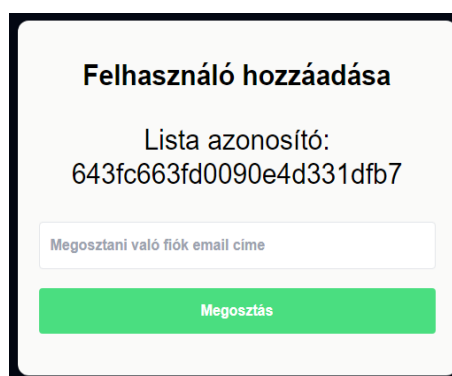


17. ábra Listák

funkcióval bármikor és bárhol hozzá lehet férni a listákhoz, és a felhasználók könnyedén megváltoztathatják a lista tartalmát, ha szükséges. Minden módosítás nyomon követhető, így a felhasználók mindig naprakész információkat láthatnak a listájukról. Összességében a Listák funkció lehetővé teszi a felhasználók számára, hogy hatékonyan és szervezeten bevásárlásukat, és minden más tevékenységüket, és egyszerűen kezeljék azokat az online felületen keresztül.

2.2.7. Felhasználó listához adás

A felhasználók számára rendkívül egyszerű a listájukhoz új embereket hozzáadni. Ehhez csak meg kell nyomni a "Megosztás" gombot, ami egy felületet nyit meg, ahol a felhasználók hozzáadhatják a kívánt személyeket az e-mail címük megadásával. Ezután csak meg kell nyomni a "Megosztás" gombot, és az új személyek hozzáadódnak a listához. A szép az



18. ábra Felhasználó hozzáadása

egészben, hogy a lista bármilyen számú emberrel bővíthető, és mindegyikük hozzájárulhat a lista módosításához, új elemek hozzáadásához és törléséhez. Mindazonáltal, fontos megjegyezni, hogy csak regisztrált felhasználók adhatók hozzá a listához, és csak ők tudják módosítani a listát. Ezen felül, azok a felhasználók, akiket

hozzáadtak a listához, képesek lesznek megnézni és kezelni azokat az elemeket, amelyeket hozzáadnak a listához, ami nagy előnyt jelent a közös bevásárlás vagy teendőik szervezésében.

2.2.8 Lista módosítás

A lista módosítása szintén a lehető legfelhasználóbarátabban készült. A módosítás opciót a lista alján található "Módosítás" gomb alatt lehet elérni. Ezután lehetőség van a lista nevének és tartalmának módosítására, valamint a tartalom törlésére is. A módosítások mentése után azok láthatóvá válnak a többi felhasználó számára is, akik hozzá lettek adva a listához. Ezzel biztosítva van, hogy az összes felhasználó naprakész információkkal rendelkezik a listáról.



19. ábra Lista módosítása

Ha bármilyen probléma merül fel a lista módosítása során, a felhasználók mindig visszaléphetnek az eredeti verzióhoz, és újra megpróbálhatják a módosításokat. Ezzel biztosítva van, hogy a lista mindig pontos és naprakész marad.

2.2.9. Lista törlés

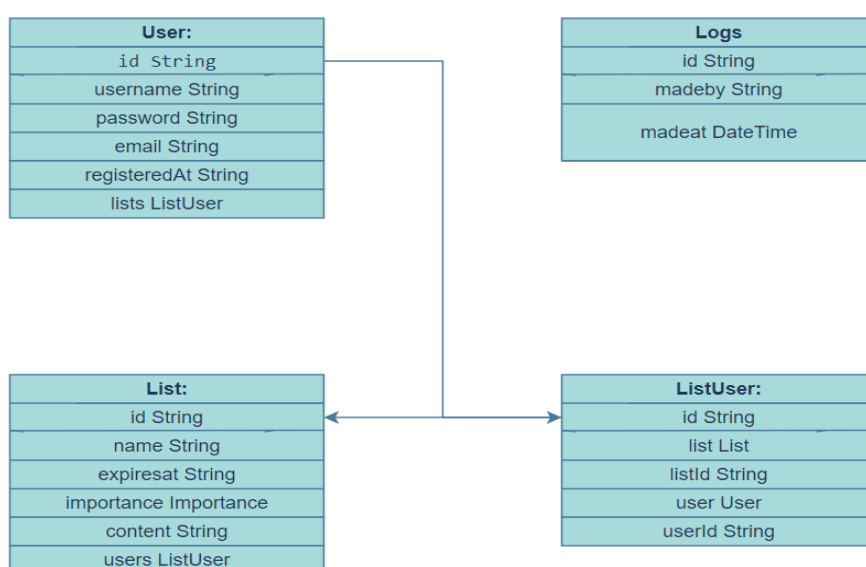
Az "Törlés" gomb az adott lista törlését eredményezi. Fontos megjegyezni, hogy ez a művelet visszavonhatatlan, és minden listaelem és minden hozzáadott felhasználótól véglegesen eltávolításra kerül. Ezért javasolt először biztosítani magunkat arról, hogy tényleg szeretnénk-e törölni a listát, és ha igen, akkor ismét megerősíteni a törlési szándékunkat.



20. ábra Lista törlése

2.3 Adatbázis

Az adatmodell, amelyet a fenti szöveg említ, egy olyan alkalmazás által használt struktúra, amely MongoDB adatbázissal kommunikál. A kommunikáció a Prisma által biztosított kliens oldali TypeScript API-n keresztül történik. Az adatmodell Object Document Mapping (ODM) tervezési mintát alkalmaz, ami lehetővé teszi az alkalmazás számára, hogy objektumokkal dolgozzon az adatokkal, amelyeknek közvetlen megfeleltetése van a MongoDB dokumentumokkal. Az adatmodell leírja a felhasználók, listák, lista-felhasználók és naplók struktúráját.



21. ábra Adatbázis kapcsolatok

A felhasználók modellje a felhasználók személyes adatait tárolja, beleértve a felhasználónevet, a jelszót, az e-mail címet és a regisztráció dátumát. A listák modellje tartalmazza a listák leírásait, beleértve a nevüket, a lejárat idejüket és a tartalmukat, valamint az általuk használt felhasználókat. A lista fontosságát az Importance felsorolás segítségével lehet beállítani. A ListUser modelle leírja, hogy mely felhasználók mely listákat használják. A Logs modellje pedig az alkalmazás eseményeit rögzíti, beleértve azok időpontját és azt, hogy melyik szereplő hozta létre azokat.

Az adatmodell implementálása hatékony kommunikációt tesz lehetővé az adatbázissal, egyszerűbbé teszi a különböző lekérdezések kezelését, és biztonságosabbá teszi az alkalmazást. Az adatmodell kódjában megtalálhatók az objektumok és a kapcsolatok

leírói, az enum-ok, és az adatbázis kapcsolatot biztosító beállítások, mint például a MongoDB adatbázis eléréséhez szükséges URL.

3. Tesztelés

3.1 Bejelentkezés teszt

Cél: A felhasználó bejelentkezik a Shopperbe.

Előfeltételek: A felhasználónak rendelkeznie kell érvényes felhasználói fiókkal.

Teszt adatok:

Felhasználónév: <a teszt felhasználóneve>

Jelszó: <a teszt jelszava>

Lépés 1: A felhasználó navigál a bejelentkező oldalra.

Lépés 2: A felhasználó megadja a teszt felhasználónevét és jelszavát az erre a célra kialakított mezőkbe.

Lépés 3: A felhasználó kattint a bejelentkezés gombra.

Lépés 4: A rendszer ellenőrzi a felhasználó által megadott adatokat, majd ha azok megfelelőek, belépteti a felhasználót a rendszerbe.

Lépés 5: Ha az adatok nem megfelelőek, akkor a rendszer visszajelez a felhasználónak a hibás adatokról, és lehetővé teszi az adatok újbóli megadását.

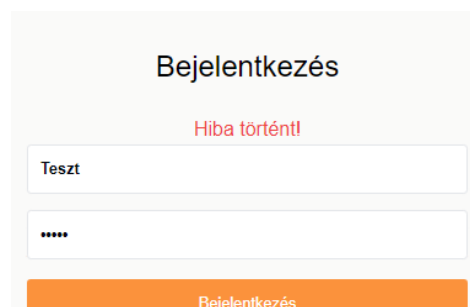
Teszteredmények:

A felhasználó sikeresen belép a rendszerbe, ha a megadott adatok érvényesek.

A rendszer visszajelez a felhasználónak, ha az általa megadott adatok nem érvényesek.

Hibajelenségek:

A felhasználó nem tud bejelentkezni a rendszerbe, ha az általa megadott adatok nem megfelelőek.



Bejelentkezés

Hiba történt!

Teszt

Bejelentkezés

22. ábra Bejelentkezési teszt

3.2 Regisztráció teszt

Cél: A felhasználó regisztrálja magát a Shopperbe.

Előfeltételek: A felhasználónak nincs még létrehozott fiókja a Shopperben.

Teszt adatok:

Felhasználónév: <a teszt felhasználóneve>

Jelszó: <a teszt jelszava>

E-mail cím: <a teszt e-mail címe>

Tesztlépések:

Lépés 1: A felhasználó navigál a regisztrációs oldalra.

23. ábra Hibás az email üzenet

Lépés 2: A felhasználó kitölti a regisztrációs űrlapot a teszt adatokkal, beleértve a felhasználónevet, jelszót és e-mail címet.

Lépés 3: A felhasználó kattint a regisztráció gombra.

teszt&teszt.com

24. ábra Regisztráció email teszt

Lépés 4: A rendszer ellenőrzi a felhasználó által

megadott adatokat, majd ha azok megfelelőek, regisztrálja a felhasználót a rendszerbe.

Lépés 5: A rendszer visszajelez a felhasználónak a sikeres regisztrációról, és lehetővé teszi a belépést az újonnan létrehozott fiókkal.

Teszteredmények:

A felhasználó sikeresen regisztrál a rendszerbe, ha a megadott adatok érvényesek.

A rendszer visszajelez a felhasználónak, ha az általa megadott adatok nem megfelelőek.

Hibajelenségek:

A felhasználó nem tud regisztrálni a rendszerbe, ha az általa megadott e mail cím érvénytelen.

A felhasználó nem tud regisztrálni a rendszerbe, ha az általa megadott jelszó nem egyezik.

A felhasználó nem tud regisztrálni a rendszerbe, ha az általa megadott jelszó nincs 8 karakter.

Regisztráció

Nem valós email cím!

Regisztráció

25. ábra 2 jelszó nem egyezik hibaüzenet

.....

....

26. ábra Hibás jelszó beírása

Regisztráció

A jelszónak minimum 8 karakter hosszúnak kell lennie!

27. ábra Hibajelző üzenet 8karakter

3.3 Lista teszt

3.3.1 Lista létrehozás

Cél: A felhasználó létrehoz egy új listát a Shoppingrebe.

Előfeltételek: A felhasználóak van érvényes profilja.

Teszt adatok:

Lista címe: <teszt lista címe>

Lista adatok: <teszt lista adatok>

Tesztlépések:

Lépés 1: A felhasználó navigál az új lista létrehozásához szolgáló oldalra.

Lépés 2: A felhasználó megadja a lista címét a megfelelő mezőbe.

Lépés 3: A felhasználó megadja a lista pontjait a megfelelő mezőbe.

Lépés 4: A felhasználó kattint a "add" gombra.

Lépés 5: A felhasználó kattint a "Hozzáadás" gombra.

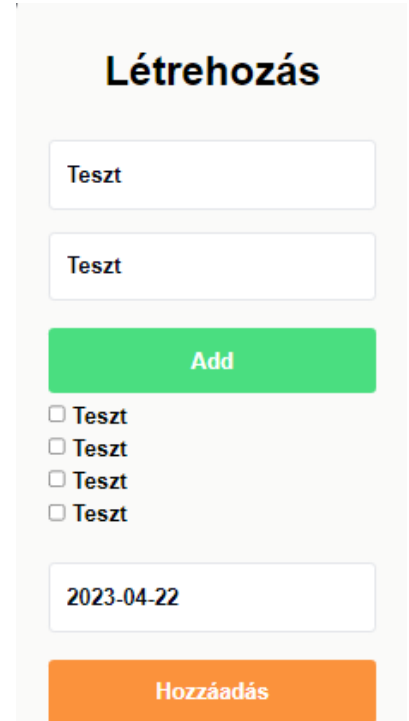
Lépés 6: A rendszer létrehozza az új listát.

Teszteredmények:

Az új lista sikeresen létrejön, ha a felhasználó rendszer megjeleníti az új lista.

Hibajelenségek:

A rendszer nem hozza létre az új listát, ha a létrehozási folyamat hibásan történt.



The screenshot shows a web form titled "Létrehozás" (Create). It contains two text input fields, both labeled "Teszt". Below the second field is a green button labeled "Add". Underneath the "Add" button are four radio buttons, each labeled "Teszt". Below the radio buttons is a date input field showing "2023-04-22". At the bottom of the form is an orange button labeled "Hozzáadás" (Add).

28. ábra Lista tesztelése

3.3.2 Megosztás

Cél: A felhasználó megoszt egy listát más felhasználókkal.

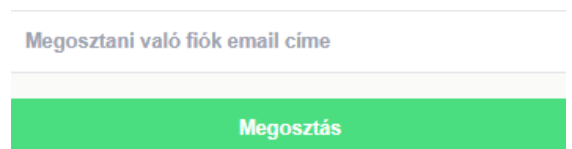
Előfeltételek: A felhasználónak létrehozott listája van a Shopingerben, és más felhasználókkal van kapcsolata.

Teszt adatok:

Címzett felhasználók: <a teszt felhasználók email cím>

Tesztlépések:

Lépés 1: A felhasználó navigál a listájához.



The screenshot shows a web form titled "Megosztás" (Share). It has a text input field labeled "Megosztani való fiók email címe" (Email address of the account to share with). Below the input field is a green button labeled "Megosztás" (Share).

29. ábra Megosztás tesztelése

Lépés 2: A felhasználó kiválasztja a megosztandó listát.

Lépés 3: A felhasználó kattint a "megosztás" gombra.

Lépés 4: A felhasználó megadja a címzett felhasználó email címét.

Lépés 5: A felhasználó kattint a "megosztás" gombra.

Lépés 6: A címzett felhasználók láthatják és szerkeszthetik a megosztott listát.

Teszteredmények:

A felhasználó sikeresen megosztja a listát, ha a címzett felhasználók hozzáférnek és szerkeszthetik azt.

A címzett felhasználók nem férnek hozzá a megosztott listához, ha a megosztási folyamat hibásan történt.

Hibajelenségek:

A felhasználó nem tudja megosztani a listát, ha a megadott email cím hibás.

3.3.3 Módosítás

Cél: A felhasználó módosít egy meglévő listát a Shopperben.

Előfeltételek: A felhasználónak jogosultsága van a lista módosítására, és az adott lista már létezik az adatbázisban.

Teszt adatok:

Módosított lista: <módosított lista>

Tesztlépések:

Lépés 1: A felhasználó navigál az adott lista oldalára.

Lépés 2: A felhasználó kattint a "Módosítás" gombra.

Lépés 3: A felhasználó módosítja a lista címét a megfelelő mezőben.

Lépés 4: A felhasználó módosítja a lista adatokat a megfelelő mezőben.

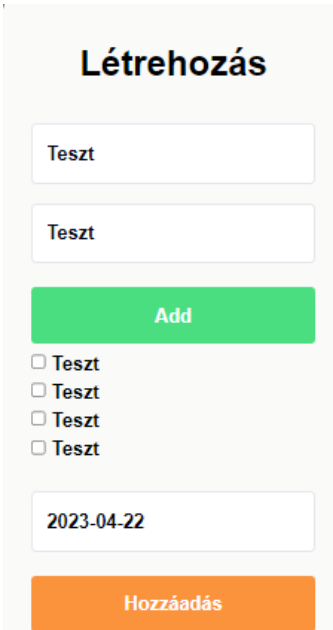
Lépés 5: A felhasználó kattint a "Mentés" gombra.

Lépés 6: A rendszer frissíti az adatbázist a módosított listával.

Teszteredmények:

A lista sikeresen módosul, ha a felhasználó visszajut az adott lista oldalára, és az új lista címe és tartalma megfelel a módosított adatoknak.

A lista nem módosul, ha a felhasználó nem nyom rá a mentésre.



30. ábra Lista módosítása

Hibajelenségek:

A rendszer nem frissíti az adatbázist, ha a módosítási folyamat hibásan történt.

3.3.4 Lista törlése

Cél: A felhasználó törli egy meglévő listát a Soppingerben.

Előfeltételek: A felhasználónak jogosultsága van a lista törlésér, és az adott lista már létezik az adatbázisban.

Teszt adatok:

Törölni kívánt lista címe: <teszt lista címe>

Tesztlépések:

Lépés 1: A felhasználó navigál az adott lista oldalára.

Lépés 2: A felhasználó kattint a "Törlés" gombra.

Lépés 3: A rendszer megjeleníti a megerősítő üzenetet.

Lépés 4: A felhasználó kattint a "Törlés" gombra a megerősítő üzeneten.

Lépés 5: A rendszer frissíti az adatbázist a törlési folyamat végrehajtásával.

Teszteredmények:

A lista sikeresen törlődik, ha a felhasználó a lépések végrehajtása után visszajut az előző oldalra, és a törölni kívánt lista már nem jelenik meg a listák között.

A lista nem törlődik, ha a felhasználó a megerősítő üzenet után nem kattint a "Törlés" gombra.

A lista nem törlődik, ha a felhasználó a megerősítő üzenetben meggondolja magát, és nem kattint a "Törlés" gombra.

Hibajelenségek:

A lista nem törlődik, ha a rendszer hibát észlel az adatbázis frissítésekor.



31. ábra Lista törlés

3.4 Menü

Cél: A felhasználó használja a menüt a rendszerben lévő lista navigációhoz, új lista hozzáadásához, kijelentkezéshez és a menü bezárásához.

Előfeltételek: A felhasználónak be kell jelentkeznie az alkalmazásba.

Teszt adatok: Nincs szükség tesztadatokra.

Tesztlépések:

Lépés 1: A felhasználó kattint a menü ikonra.

Lépés 2: A rendszer megjeleníti a menüt a lista navigáció, az új lista hozzáadása, a kijelentkezés és a bezárása opciókkal.

Lépés 3: A felhasználó kattint a "Lista navigáció" opcióra.

Lépés 4: A rendszer átirányítja a felhasználót a listák oldalra.

Lépés 5: A felhasználó kattint a menü ikonra.

Lépés 6: A felhasználó kattint az "Új lista hozzáadása" opcióra.

Lépés 7: A rendszer átirányítja a felhasználót az új lista létrehozása oldalra.

Lépés 8: A felhasználó kattint a menü ikonra.

Lépés 9: A felhasználó kattint a "Kijelentkezés" opcióra.

Lépés 10: A rendszer megjeleníti a megerősítő üzenetet.

Lépés 11: A rendszer kijelentkezteti a felhasználót, és átirányítja a felhasználót a bejelentkező oldalra.

Lépés 12: A felhasználó kattint a menü ikonra.

Lépés 13: A felhasználó kattint a "Bezárás" opcióra.

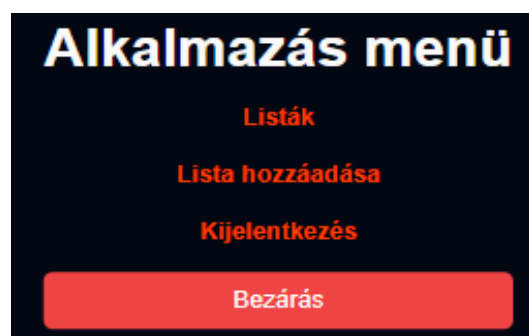
Lépés 14: A rendszer bezárja a menüt.

Teszteredmények:

A lista navigáció sikeresen működik, ha a felhasználó a lépések végrehajtása után visszajut a listák oldalra.

Az új lista létrehozása sikeresen működik, ha a felhasználó a lépések végrehajtása után visszajut az új lista létrehozása oldalra.

A kijelentkezés sikeres, ha a felhasználó a lépések végrehajtása után átirányítást kap az Otthon oldalra.



32. ábra Menü

Összegzés

Az online lista generátor, amely Svelte és TypeScript technológiákat használ, lehetővé teszi a felhasználók számára, hogy létrehozzanak és kezeljenek listákat. A felhasználók regisztrálhatnak vagy bejelentkezhetnek a Facebook használatával. A regisztráció során a felhasználóknak meg kell adniuk nevüket, e-mail címüket és jelszavukat, valamint meg kell erősíteniük a jelszavukat.

Az alkalmazás backendje TypeScript backend technológia felhasználásával lett készítve. A backend felelős az adatbázis-kezelésért, a biztonságért, a felhasználói adatok ellenőrzéséért, valamint az API-k és a ruterek kezeléséért. A backend és a frontend közötti kommunikáció általában RESTful API-kon keresztül történik.

A frontend részt az Svelte keretrendszer használatával lehet elkészíteni. Az Svelte egy modern JavaScript keretrendszer, amely lehetővé teszi a fejlesztők számára, hogy hatékonyan és gyorsan fejlesszenek web alkalmazásokat. Az Svelte az adatok megjelenítéséhez komponenseket használ, amelyek automatikusan újra renderelődnek, amikor az adatok megváltoznak.

A frontend fejlesztés során a TypeScriptet használhatjuk, hogy statikus típus ellenőrzést végezzünk a kódban. A TypeScript növeli az alkalmazás biztonságát és megbízhatóságát, mivel elősegíti a gyakori hibák megelőzését. Emellett javítja a fejlesztői élményt az IntelliSense segítségével, amely javaslatokat ad a kódolás során, és javítja az olvashatóságot és dokumentálhatóságot.

Az adatok tárolása az alkalmazásban lehet localStorage, indexedDB vagy egy másik adatbázis-kezelő technológia felhasználásával. Az adatokat a backend segítségével lehet menteni és lekérdezni.

Az alkalmazás tervezésekor figyelembe kell venni az alkalmazás skálázhatóságát, biztonságát és felhasználói élményét. Az alkalmazásnak biztonságosnak kell lennie, és meg kell akadályozni az illetéktelen hozzáférést. Az alkalmazásnak lehetővé kell tennie a felhasználók számára, hogy könnyen létrehozzanak és kezeljenek listákat, és hogy bármikor módosíts

Ábrajegyzék

1. ábra HTML logo.....	9
2. ábra CSS logo.....	10
3. ábra Svelte logo.....	10
4. ábra Typescript.....	11
5. ábra JavaScript logo.....	12
6. ábra Prisma logo.....	12
7. ábra Regisztrációs felület.....	14
8. ábra Bejelentkezési felület.....	14
9. ábra Üdvözlő.....	15
10. ábra Személyes üdvözlés.....	15
11. ábra Bemutató.....	15
12. ábra Menü ikon.....	16
13. ábra Menü.....	16
14. ábra Otthoni listák.....	16
15. ábra Lista.....	17
16. ábra Lista létrehozási panel.....	17
17. ábra Listák.....	18
18. ábra Felhasználó hozzáadása.....	18
19. ábra Lista módosítása.....	19
20. ábra Lista törlése.....	19
21. ábra Adatbázis kapcsolatok.....	20
22. ábra Bejelentkezési teszt.....	22
23. ábra Hiba az email üzenet.....	23
24. ábra Regisztráció email teszt.....	23
25. ábra 2 jelszó nem egyezik hibaüzenet.....	23
26. ábra Hiba jelszó beírása.....	23
27. ábra Hibajelző üzenet 8karakter.....	23
29. ábra Lista tesztelése.....	24
30. ábra Megosztás tesztelése.....	24
31. ábra Lista módosítása.....	25
32. ábra Lista törlés.....	26
33. ábra Menü.....	27

Irodalmijegyzék

1. <https://svelte.dev/>
2. <https://www.typescriptlang.org/>
3. <https://www.mongodb.com/>
4. <https://www.prisma.io/>
5. <https://en.wikipedia.org/wiki/JavaScript>
6. <https://en.wikipedia.org/wiki/CSS>

Mellékletek

<https://github.com/NTSZKI22/SzakdogaBevasarlo>