

Baranya Vármegyei Szakképzési Centrum

Simonyi Károly Technikum és Szakképző Iskola

Szakma megnevezése: Szoftverfejlesztő és -tesztelő

A szakma azonosító száma: 5 0613 12 03

Vizsgaremek

Mátyás Csárda

Készítették: Lakatos Károly

Németh Kristóf Zsolt

Takács Krisztián

Pécs

2025

2025 EREDETISÉGI NYILATKOZAT

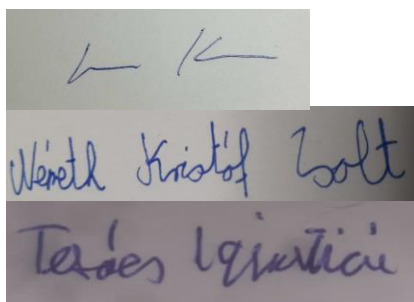
Alulírottak: Lakatos Károly, Takács Krisztián, Németh Kristóf Zsolt

A Baranya Vármegyei SzC Simonyi Károly Technikum és Szakképző Iskola, Szoftverfejlesztő és tesztelő végzős tanulói, büntetőjogi és fegyelmi felelősségünk tudatában nyilatkozunk és aláírásunkkal igazoljuk, hogy a(z)

Mátyás Csárda című vizsgaremek saját, önálló munkánk, az abban hivatkozott szakirodalom felhasználása a forráskezelés szabályai szerint történt. Tudomásul vesszük, hogy vizsgaremek esetén plágiumnak számít:

- szószerinti idézet közlése idézőjel és hivatkozás megjelölése nélkül;
- tartalmi idézet hivatkozás megjelölése nélkül;
- más publikált gondolatainak saját gondolatként való feltüntetése.

Alulírott kijelentjük, hogy a plágium fogalmát megismertük, és tudomásul vesszük, hogy plágium esetén a vizsgaremekünk visszautasításra kerül.



The image shows three lines of handwritten signatures in blue ink. The first line contains two simple, stylized marks. The second line reads 'Németh Kristóf Zsolt'. The third line reads 'Takács Krisztián'.

Tanulók aláírása

Tartalomjegyzék

| | |
|---|----|
| Bevezető | 4 |
| A feladat ismertetése..... | 4 |
| A választott téma indoklása..... | 4 |
| Munkamegosztás a csapatban..... | 4 |
| Szerzett tapasztalatok..... | 5 |
| Felhasználói dokumentáció | 6 |
| A program általános specifikációja | 6 |
| Rendszerkövetelmények..... | 6 |
| Program részletes telepítése és elindítása | 8 |
| A program használatának részletes leírása | 11 |
| Fejlesztői dokumentáció | 21 |
| Alkalmazott fejlesztői eszközök | 21 |
| Tervezési módszerek..... | 24 |
| Adatmodell leírása | 27 |
| Részletes feladat-specifikáció és algoritmusok..... | 33 |
| Tesztelési dokumentáció | 38 |
| Továbbfejlesztői lehetőségek | 41 |
| Források, segítség | 41 |

Bevezető

A feladat ismertetése

Projektünk célja egy olyan felhasználóbarát webes alkalmazás fejlesztése volt, amely lehetőséget biztosít az éttermi étlap áttekintésére, ételek rendelésére, valamint az adminisztrációs feladatok gyors és egyszerű kezelésére az admin felhasználó számára. Kiemelt figyelmet fordítottunk arra, hogy a felület reszponzív legyen, így mobiltelefonon, tableten és asztali számítógépen egyaránt zökkenőmentes felhasználói élményt nyújtson. A kialakítás során olyan technológiákat választottunk, amelyek biztosítják az adatok biztonságos tárolását és kezelését. Ezen kívül a rendszer úgy lett felépítve, hogy az a jövőben könnyedén továbbfejleszthető és bővíthető legyen. Célunk az volt, hogy mind a vendégek, mind az adminisztrátor számára kényelmes és hatékony megoldást kínáljunk. A projekt során végig szem előtt tartottuk az átláthatóságot, a használhatóságot és a biztonságot.

A választott téma indoklása

Az éttermek digitális platformokon történő jelenléte napjainkban egyre fontosabbá válik, és egyre nagyobb igény mutatkozik ezekre a szolgáltatásokra. Az alkalmazás fejlesztése során nemcsak technikai ismereteket szereztünk, hanem arra is figyeltünk, hogy a piaci igényekre és felhasználói elvárásokra is válaszoljunk. Ezzel biztosítva, hogy az alkalmazás releváns és versenyképes maradjon a gyorsan változó digitális környezetben.

Kinek készült?

A magyaros hazai ízeket kedvelő vendégek számára készült a Mátyás csárda weboldala, akik értékelik a tradicionális ételeket, a házas vendégszeretetet, vagy szívesebben rendelnek otthonuk kényelméből, hogy megkóstolják a magyar konyha ízvilágát.

Munkamegosztás a csapatban

Lakatos Károly: Backend és frontend fejlesztés, adatbázis-tervezés és dokumentálás.

Németh Kristóf Zsolt: Dizájn tervezése, képek és információk gyűjtése, illetve dokumentálás.

Takács Krisztián: Dizájn, adatbázis fejlesztése, tervezése és dokumentálás.

Szerzett tapasztalatok

Kommunikációs eszközök hatékony használata

A **GitHub** lehetőséget biztosított a kódverziók nyomon követésére és egyszerű megosztására, így hatékonyan tudtunk együtt dolgozni a fejlesztés során. A **Discord** pedig folyamatos kommunikációt és gyors fájlmegosztást tett lehetővé, támogatva az együttműködést és az információáramlást.

Problémamegoldási készségek

Számos technikai probléma megoldása során értékes tapasztalatokat szereztünk a hibakeresésben és a problémák gyors kezelésében. Például, amikor adatbázis-kapcsolati hibákkal vagy a reszponzív dizájn elcsúszásával találkoztunk, megtanultuk gyorsan beazonosítani a hiba forrását. Fokozatosan kialakítottunk egy hatékony problémamegoldási rutint. A közös munkafolyamatok során fejlesztettük az elemző gondolkodásmódunkat és a technikai tudásunkat. Egyre magabiztosabban nyúltunk hozzá az eszközökhöz és megoldásokhoz, amelyekkel korábban még bizonytalanok voltunk. Ezek a kihívások segítettek abban, hogy csapatként is jobban együttműködjünk. Felismertük, mennyire fontos a gyors reakció és a pontos kommunikáció a hatékony hibakezeléshez. Mindezek eredményeként jelentősen fejlődött a problémamegoldó készségünk és a technikai felkészültségünk.

Programozási készségek

A PHP használata során mélyebben megértettük az objektumorientált programozás alapelveit és előnyeit. Megtanultuk, hogyan építsünk fel osztályokat, és hogyan használjuk ezeket a kód strukturáltabbá tételére. Rájöttünk, hogy az osztályok segítenek az adatok logikus kezelésében, valamint abban, hogy a funkciókat jól elkülönítve, átláthatóan szervezzük. Tapasztalatot szereztünk abban is, hogyan tegyük a kódot újra felhasználhatóvá és könnyebben karbantarthatóvá. A moduláris felépítésnek köszönhetően egyszerűbbé vált a fejlesztés és a hibakeresés is. Felismertük, hogy az objektumorientált megközelítés segít a nagyobb projektek átlátásában és kezelésében. A PHP gyakorlati alkalmazása közben egyre magabiztosabbá váltunk ebben a szemléletmódban. Ezáltal hatékonyabban tudtunk együtt dolgozni és szervezettebben fejleszteni.

Felhasználói dokumentáció

A program általános specifikációja

Felhasználói bejelentkezés és regisztráció

A rendszer lehetővé teszi új felhasználók regisztrációját, valamint a már meglévő profilokkal történő bejelentkezést is. Ez a funkcionalitás alapot ad a személyre szabott felhasználói élmény kialakításához, például a rendelések nyomon követéséhez vagy korábbi adatok eléréséhez. A biztonság kiemelt szerepet kapott, ezért a jelszavakat titkosított formában tároljuk. Ezzel megóvjuk a felhasználók adatait az illetéktelen hozzáféréstől. Valamint a bejelentkezési rendszer megbízható és könnyen használható felületet biztosít minden eszközön.

Étlap

Minden ételhez tartozik egy rövid, informatív leírás, amely segíti a választást, valamint egy jól látható kép, ami vizuálisan is bemutatja az ételt. Emellett az ár is egyértelműen feltüntetésre kerül, így a felhasználók könnyedén áttekinthetik a kínálatot. Ez a megjelenítés egyszerre esztétikus és praktikus, hozzájárulva a kellemes böngészési élményhez.

Kosárkezelés

A felhasználók összeállíthatják rendeléseiket a kosár segítségével, ahol láthatják az összes hozzáadott ételt, annak árát és a rendelt mennyiséget. A kosár tartalma menet közben módosítható, például tételek törlésével, hozzáadásával vagy mennyiségek változtatásával. A rendelési adatok véglegesítése előtt, van még esély a rendelés mód

Rendelés leadása

A felhasználóknak a rendelés véglegesítésekor meg kell adniuk a szállítási címet és a telefonszámot, valamint kiválaszthatják a számukra megfelelő fizetési módot. Jelenleg az alkalmazás támogatja a készpénzes fizetést, amely egyszerű és gyors megoldást kínál. A rendszer kialakítása azonban lehetővé teszi, hogy a jövőben további fizetési lehetőségekkel, például bankkártyás fizetéssel is bővüljön. Ez rugalmasabbá teszi a rendelési folyamatot és alkalmazkodik a felhasználói igényekhez.

Profil megtekintése

A profil oldalon a felhasználók könnyedén megtekinthetik személyes adataikat, valamint hozzáférhetnek korábbi rendeléseikhez. A rendszer minden egyes rendeléshez részletes információkat biztosít, mint az étkezés összára, a kiválasztott ételek nevei és a rendelés dátuma. Ez lehetővé teszi, hogy a felhasználók gyorsan áttekintsék előző vásárlásaikat, és egyszerűen újra rendeljenek kedvenc ételeikből. Az oldal tiszta és áttekinthető dizájnja segíti a könnyű navigációt és a gyors információkeresést.

Rendszerkövetelmények

A weboldal futtatásához szükséges program

Google Chrome

A következő rendszerkövetelmények szükségesek az alkalmazás telepítéséhez és futtatásához:

Operációs rendszer: Windows 10 vagy újabb
Processzor: Intel Pentium 4 vagy újabb (SSE3 támogatással)
Memória (RAM): Minimum 2 GB (ajánlott 4 GB)
Tárhely: Legalább 200 MB szabad lemezterület
Kijelző: Minimum 1024x768 képernyőfelbontás
Internetkapcsolat: Szükséges a telepítéshez és a szinkronizáláshoz

XAMPP

A következő rendszerkövetelmények szükségesek az alkalmazás telepítéséhez és futtatásához:

Operációs rendszer: Windows 10 vagy 11 (32 vagy 64 bites)
Processzor: Legalább 1.5 GHz-es többmagos processzor
Memória (RAM): Minimum 2 GB (ajánlott 4 GB vagy több)
Tárhely: Legalább 1 GB szabad lemezterület
Jogosultság: Rendszergazdai jogosultság szükséges a telepítéshez

MySQL

A következő rendszerkövetelmények szükségesek az alkalmazás telepítéséhez és futtatásához:

Operációs rendszer: Windows 10, 11, Windows Server 2016, 2019, 2022
Processzor: Legalább 1.5 GHz-es többmagos processzor
Memória (RAM): Minimum 1 GB (ajánlott 4 GB vagy több, nagyobb adatbázisokhoz)
Tárhely: Minimum 200 MB szabad lemezterület

Ezek a követelmények biztosítják majd az alkalmazások zökkenőmentes működését különböző operációs rendszereken.

Program részletes telepítése és elindítása

Nyissa meg a GitHub weboldalt: Látogasson el a GitHub hivatalos weboldalára az alábbi linken: <https://github.com/>

Regisztráljon egy új fiókot:

Ha még nincs GitHub fiókja, kattintson a „Sign up” vagy „Regisztráció” gombra.

Töltse ki a szükséges információkat, mint például felhasználónevet, e-mail címet és jelszót. Kövesse az utasításokat a fiók létrehozásához és a biztonsági ellenőrzések befejezéséhez.

Jelentkezzen be a frissen létrehozott fiókjával:

Miután sikeresen regisztrált, kattintson a „Sign in” vagy „Bejelentkezés” gombra. Adja meg felhasználónevét és jelszavát a bejelentkezéshez.

Sign up to GitHub

Email*

Password*

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your country*

Hungary

For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Email preferences

☐ Receive occasional product updates and announcements

Continue >

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.



Sign in to GitHub

Username or email address

Password

[Forgot password?](#)

Sign in

[Sign in with a passkey](#)

[New to GitHub? Create an account](#)

Fájlok letöltése

Lépjen be a GitHub fiókjába: Használja a regisztrált felhasználónevét és jelszavát a bejelentkezéshez. Nyissa meg a fájlok tartalmazó linket: Az alábbi linkre kattintva elérheti a fájlokat: Link: <https://github.com/Karoly150/Projekt-munka/tree/main>

Telepítse a szükséges adatbázist: A letöltött fájlok között megtalálja a matyas_csarda.sql fájlt. Töltse le a vizsga mappát: A mappában megtalálhatóak a weboldal működéséhez szükséges összes fájl.

Kattintson a "Download ZIP" gombra, hogy letöltse a mappát és a fájlokat számítógépére.

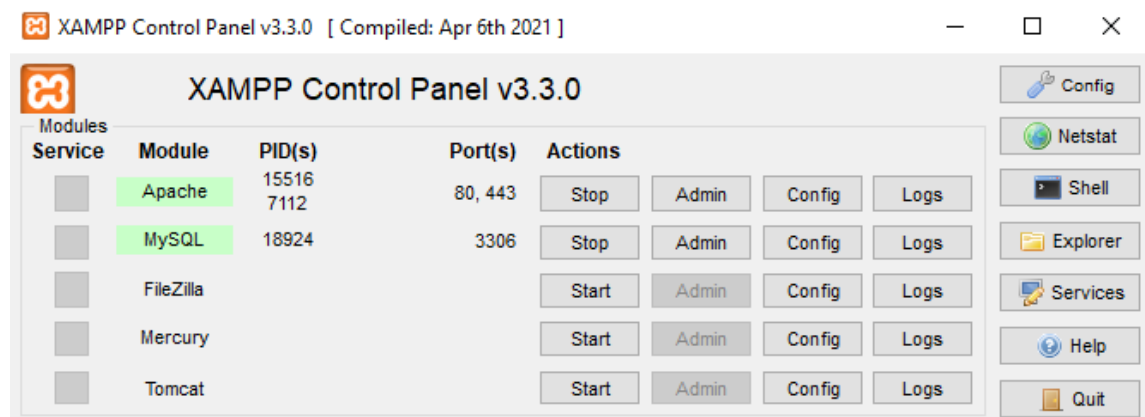
| | |
|--|-----------------------------|
| Vizsga | Add files via upload |
| Baranya Megyei Szakképzési Centrum.pdf | Add files via upload |
| Németh_Kristóf_Zsolt_13c_dizajn_css_és_javasc... | Dizájnok (css + javascript) |
| README.md | Update README.md |
| matyas_csarda (1).sql | Add files via upload |

XAMPP letöltése és telepítése

Nyissa meg a XAMPP hivatalos weboldalát az alábbi linken: <https://www.apachefriends.org/download.html>, ahol elérheti a különböző operációs rendszerekhez (Windows, Linux vagy macOS) megfelelő verziókat. Válassza ki az operációs rendszerének megfelelő verziót, majd töltsse le a telepítőt. Miután a fájl letöltődött, futtassa a telepítőt, és kövesse a telepítési útmutatót. A telepítés során fontos, hogy engedélyezze az „Apache” és „MySQL” modulokat, mivel ezek szükségesek a weboldal futtatásához és az adatbázis kezeléséhez.

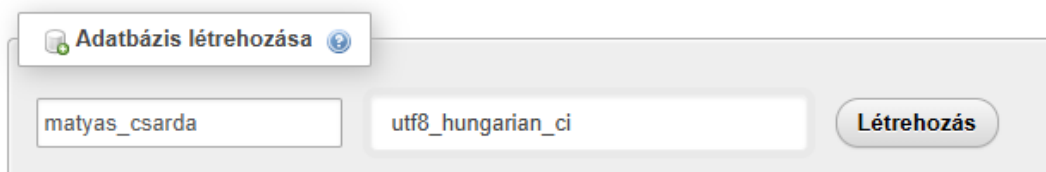
Apache és MySQL indítása

Indítsa el a XAMPP-ot, majd a vezérlőpanelen kattintson az „Apache” és „MySQL” melletti „Start” gombokra. Miután elindította őket, győződjön meg arról, hogy mindkét modul zöld színűre vált, ami azt jelzi, hogy az „Apache” és „MySQL” aktív állapotban van. Ha mindkét modul zöld színű, akkor sikeresen elindultak, és készen állnak a használatra.

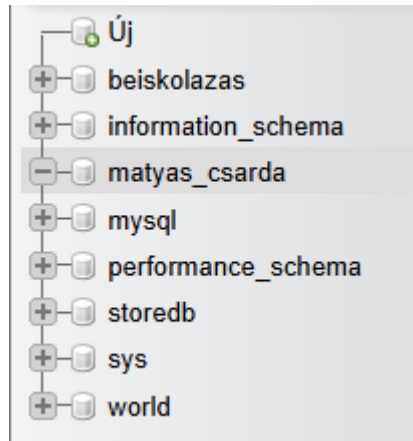


Az adatbázis létrehozása és importálása

Nyisson meg egy böngészőt, és a címsorba írja be a localhost/phpmyadmin linket, hogy elérje a phpMyAdmin felületét. Miután belépett, hozzon létre egy új adatbázist a következő névvel: matyas_csarda. Az adatbázis létrehozása után válassza az „Importálás” fület, majd kattintson a „Tallózás” gombra, hogy kiválassza a Github tárolóból letöltött matyas_csarda.sql fájlt. Miután kiválasztotta a fájlt, indítsa el az importálási folyamatot, és várja meg, amíg a fájl teljesen importálódik az adatbázisba.



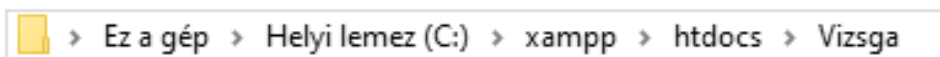
Ha a létrehozás és az importálás sikeres, akkor az adatbázisnak, matyas_csarda néven, meg kell jelennie a phpMyAdmin bal oldali paneljén. Kattintson rá, és látnia kell a benne található táblákat, amelyek a letöltött matyas_csarda.sql fájlban találhatóak. Ha a táblák is megjelennek, akkor az importálás sikeresen befejeződött, és az adatbázis készen áll a használatra.



| Tábla | Művelet | Sorok | Típus | Illesztés | Méret | Felülírás |
|---------------------------------------|--|-------|--------|--------------------|----------|-----------|
| <input type="checkbox"/> adminok | ★ Tartalom Szerkezet Keresés Beszúrás Kiűrités Eldobás | 1 | InnoDB | utf8mb4_general_ci | 16.0 KB | - |
| <input type="checkbox"/> etelek | ★ Tartalom Szerkezet Keresés Beszúrás Kiűrités Eldobás | 24 | InnoDB | utf8mb4_general_ci | 32.0 KB | - |
| <input type="checkbox"/> felhasznalok | ★ Tartalom Szerkezet Keresés Beszúrás Kiűrités Eldobás | 4 | InnoDB | utf8mb4_general_ci | 32.0 KB | - |
| <input type="checkbox"/> kosar | ★ Tartalom Szerkezet Keresés Beszúrás Kiűrités Eldobás | 1 | InnoDB | utf8mb4_general_ci | 48.0 KB | - |
| <input type="checkbox"/> rendelesek | ★ Tartalom Szerkezet Keresés Beszúrás Kiűrités Eldobás | 9 | InnoDB | utf8mb4_general_ci | 48.0 KB | - |
| 5 tábla | Összesen | 39 | InnoDB | utf8mb4_general_ci | 176.0 KB | 0 B |

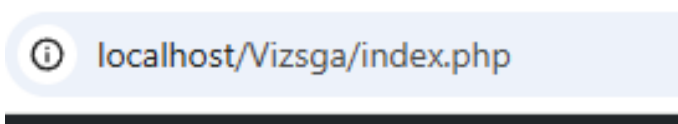
A fájlok elhelyezése a XAMPP htdocs mappájába

A korábban letöltött fájlokat másolja át a XAMPP telepítési mappájában található htdocs könyvtárba. Ehhez nyissa meg a XAMPP telepítési mappáját (általában C:\xampp\), és keresd meg a htdocs mappát. Ebbe a mappába másolja be a letöltött fájlokat, például a vizsga mappát és az összes szükséges fájlt.



A weboldal elérése böngészőből

Nyisson meg egy böngészőt, és írja be a következő címet: <http://localhost/Vizsga/index.php>. Ha minden megfelelően be van állítva, az oldal betöltődik, és hozzáférhet a programhoz. Ekkor elkezdheti a webalkalmazás használatát, és elérheti annak összes funkcióját.



A program használatának részletes leírása

Fejléc - navigáció az oldalak között

A weboldal fejlécében elhelyezett navigációs sáv lehetővé teszi az oldalak közötti könnyű váltást, miközben minden látogató számára jól láthatóan megjelenik a „Mátyás Csárda” felirat, amely hivatkozásként is működik, és a kezdőlapra irányítja vissza a felhasználót. A bejelentkezett felhasználók számára a navigációs sávban elérhető a profiloldal, az étlap, a rendelés, a kosár és a kijelentkezés menüpont, míg a nem bejelentkezett látogatók csak az étlapot, a rendelés menüpontot és a bejelentkezési lehetőséget látják. Ez a megoldás segíti a felhasználói élmény fokozását, mivel mindenki csak a számára releváns funkciókat éri el a felületen.

Bejelentkezett felhasználók számára megjelenő fejléc



Nem bejelentkezett felhasználók számára megjelenő fejléc



Kezdő oldal

A fő tartalom egy rövid bemutatkozó résszel indul, amelynek célja, hogy megismertesse az új látogatókat a Mátyás Csárda értékeivel és történetével. A bemutatkozásban olvasható egy tömör leírás az étterem filozófiájáról, valamint a kínált ételek jellegéről és minőségéről. Az oldalon megtalálhatók az elérhetőségek is, így a látogatók megismerhetik az étterem pontos helyszínét – amely térképen is megtekinthető – és a kapcsolattartáshoz szükséges információkat. Emellett részletesen feltüntetésre kerül az étterem nyitvatartási ideje, hogy a vendégek tudják, mikor érdemes ellátogatniuk. Végül, az oldal lehetőséget kínál asztalfoglalásra is, ahol a felhasználók egyszerűen és gyorsan elindíthatják a foglalási folyamatot.

Bejelentkezés

A bejelentkezés lehetőséget biztosít a felhasználók számára, hogy biztonságos módon hozzáférjenek személyes profiljukhoz, megtekinthessék korábbi rendeléseiket, valamint használni tudják az alkalmazás rendelési funkcióit. A bejelentkezési oldal minden esetben elérhető a fejlécben elhelyezett menüponton keresztül, kivéve, ha a felhasználó már be van jelentkezve.

Bejelentkezési folyamat

A bejelentkezési folyamat során a felhasználónak meg kell adnia a felhasználónevét, amelyet a regisztráció során használt, valamint a hozzá tartozó jelszót. Amennyiben az adatok helyesek, a felhasználó bejelentkezik, és teljes hozzáférést kap a számára elérhető funkciókhoz. Hibás adatok esetén a rendszer visszajelzést ad: „Hibás felhasználónév vagy jelszó.”, és lehetőséget biztosít az újbóli próbálkozásra.

Bejelentkezés

Felhasználónév

Jelszó

Bejelentkezés

[Még nincs fiókja? Regisztráljon itt.](#)

Hibás felhasználónév vagy jelszó!

Sikeres bejelentkezés esetén az alkalmazás automatikusan átirányítja a felhasználót a kezdőoldalra, ahol már hozzáférhet a bejelentkezett felhasználók számára elérhető funkciókhoz.

Regisztráció

A regisztráció az alkalmazás egyik alapvető funkciója, amely lehetővé teszi a felhasználók számára, hogy saját profilt hozzanak létre. A profil segítségével a rendszer személyre szabott élményt kínál. A regisztráció révén a felhasználók teljes hozzáférést kapnak az alkalmazás minden funkciójához, beleértve a rendelést, a kosár kezelését és a profiloldalt is.

Hol érhető el a regisztráció?

Amikor egy felhasználó a bejelentkezési oldalra navigál, az oldal nemcsak a belépés lehetőségét kínálja fel, hanem egyértelműen megjeleníti a regisztrációs opciót is azok számára, akik még nem rendelkeznek fiókkal. A bejelentkezési űrlap alatt található egy jól látható hivatkozás a következő szöveggel: „Még nincs fiókja? Regisztráljon itt.” Ez a link közvetlenül a regisztrációs oldalra irányítja a felhasználót, ahol könnyedén létrehozhatja saját profilját az alkalmazásban.

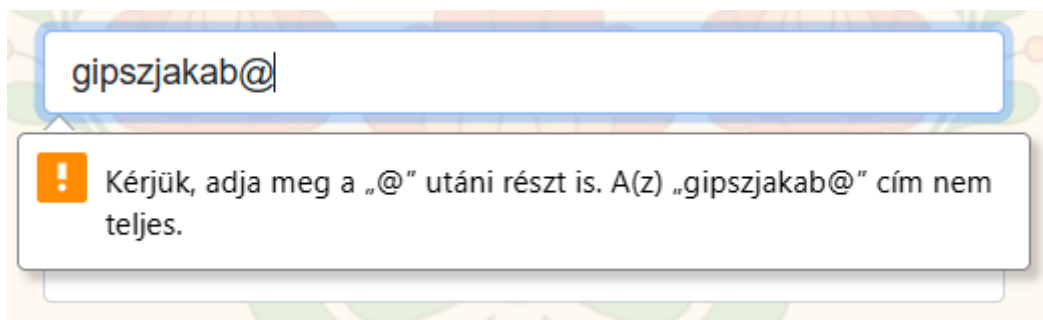
[Még nincs fiókja? Regisztráljon itt.](#)

A regisztráció menete

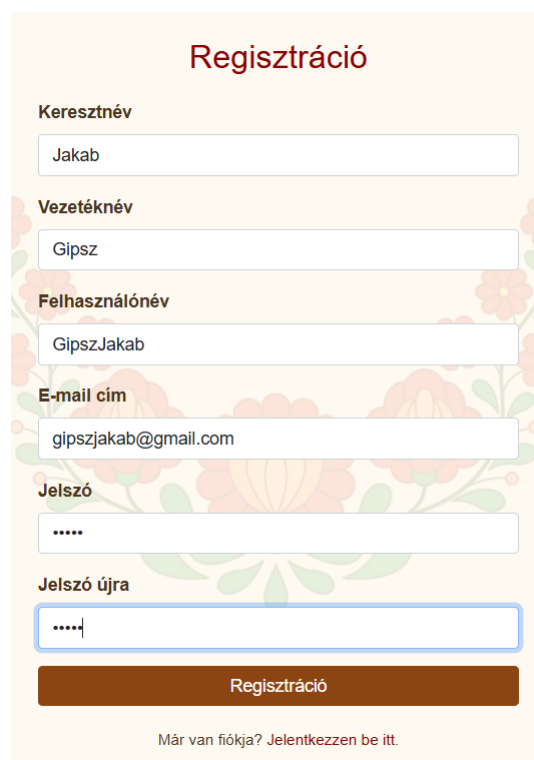
A regisztráció folyamata során a felhasználónak több adatot kell megadnia annak érdekében, hogy létrehozassa saját fiókját az alkalmazásban. A kötelező mezők között szerepel a felhasználónév, keresztnév, vezetéknév, valamint egy érvényes e-mail cím, amelyet a rendszer automatikusan ellenőriz, különös tekintettel a helyes formátumra (a „@” és a „.hu” vagy „.com” meglétére). Ezen kívül biztonságos jelszót kell megadni, amely legalább 8 karakter hosszú, és tartalmaz kis- és nagybetűt, valamint speciális karaktert. A jelszót a felhasználónak meg kell erősítenie, hogy kizárja a véletlen elírásokat.



The screenshot shows a registration form with a label "Jelszó" (Password) above a text input field. Below the field is a label "Jelszó újra" (Repeat password). A red error message box is displayed, containing an exclamation mark icon and the text: "Kérjük, töltsse ki ezt a mezőt." (Please fill out this field.).



The screenshot shows a registration form with a text input field containing "gipszjakab@". Below the field is a red error message box, containing an exclamation mark icon and the text: "Kérjük, adja meg a „@” utáni részt is. A(z) „gipszjakab@” cím nem teljes." (Please provide the part after the "@". The (z) „gipszjakab@" email address is not complete.).



The screenshot shows the full registration form titled "Regisztráció" (Registration). The form contains the following fields and labels:

- Keresztnév** (First name): Input field with "Jakab".
- Vezetéknév** (Last name): Input field with "Gipsz".
- Felhasználónév** (Username): Input field with "GipszJakab".
- E-mail cím** (Email address): Input field with "gipszjakab@gmail.com".
- Jelszó** (Password): Input field with masked characters ".....".
- Jelszó újra** (Repeat password): Input field with masked characters ".....".

Below the fields is a brown button labeled "Regisztráció". At the bottom of the form, there is a link: "Már van fiókja? Jelentkezzen be itt." (Already have an account? Log in here.).

Miután minden mezőt megfelelően kitöltött, a „Regisztráció” gombra kattintva a rendszer ellenőrzi az adatokat. Sikeres regisztráció esetén megjelenik egy visszajelzés és a felhasználó bejelentkezhet:

Sikeres regisztráció!

Köszönjük, hogy regisztrált. Most már bejelentkezhet a fiókjába.

Tovább a bejelentkezésre

” Amennyiben hiba történik, a rendszer figyelmeztető üzenetet jelenít meg.

Profil oldal

Lehetőségek és funkciók a profil oldalon

A profil oldal egyik kiemelt funkciója, hogy lehetőséget biztosít a felhasználóknak korábbi rendeléseik részletes áttekintésére. A rendszer minden rendelést dátummal lát el, így megjelenik a rendelés pontos időpontja. Emellett látható a teljes rendelési összeg, valamint a rendelt ételek listája is, amely tartalmazza az egyes ételek nevét, a hozzájuk tartozó mennyiséget, illetve a felhasználó láthatja, hogy megrendelése milyen státuszban áll. A rendelések időrendi sorrendben kerülnek megjelenítésre, mindig a legfrissebbtől indulva, ezáltal a felhasználó könnyedén visszakeresheti vásárlási előzményeit.

Korábbi rendeléseim

| Rendelési azonosító | Rendelés dátuma | Étel neve | Mennyiség | Státusz |
|---------------------|------------------|---------------------------|-----------|----------|
| 101 | 2025-04-15 06:53 | Marhapörkölt galuskával | 1 | függőben |
| 102 | 2025-04-15 06:53 | Csirkepaprikás galuskával | 1 | függőben |

A profil oldalon a felhasználók könnyedén hozzáférhetnek azokhoz a személyes adataikhoz, amelyeket a regisztráció során adtak meg. Ezek között szerepel a teljes név, a regisztrált e-mail cím, valamint a választott felhasználónév. A rendszer szándékosan nem tárol olyan adatokat, mint a lakcím vagy a telefonszám, mivel ezek az információk gyakran változhatnak, és praktikusabb, ha a felhasználó ezeket minden rendelésnél újra megadhatja. Ez a megoldás nemcsak egyszerűbbé teszi az adatkezelést, hanem hozzájárul a személyes adatok védelméhez is.

Profilom



Felhasználónév: ErvinNagy

Teljes név: Ervin Nagy

E-mail: nagyervin@gmail.com

Étlap

Az étlap oldal az alkalmazás egyik központi eleme, amely lehetővé teszi a felhasználók számára bejelentkezés nélkül, hogy részletesen megtekintsék az étterem kínálatát. Az oldal kialakításának célja, hogy az ételek átlátható és könnyen kezelhető módon jelenjenek meg, ezáltal megkönnyítve a felhasználók számára a rendelési folyamatot. Minden egyes ételhez tartozik egy jól látható kép, amely vizuálisan is vonzóvá teszi az ajánlatokat, valamint egy rövid leírás, amely az étel összetevőit ismerteti. Emellett feltüntetésre kerül az adott fogás ára is, így a felhasználók minden szükséges információt egy helyen megtalálhatnak a döntéshez.



Rendelés

A rendelés oldal az alkalmazás egyik kulcsfontosságú funkcióját látja el, hiszen itt van lehetőségük a felhasználóknak arra, hogy a korábban kiválasztott ételeket ténylegesen megrendeljék. Az oldal első lépéseként ellenőrzi, hogy a felhasználó be van-e jelentkezve, mivel a rendelési lehetőségek kizárólag regisztrált és aktív fiókkal rendelkező felhasználók számára elérhetők. Amennyiben a látogató nincs bejelentkezve, a rendszer egy figyelmeztető üzenetet jelenít meg: „A rendelés megtekintéséhez kérjük, jelentkezzen be.” Az üzenet mellett egy hivatkozás is megjelenik, amely a bejelentkezési oldalra irányítja a felhasználót, így biztosítva a gyors és egyszerű hozzáférést a szükséges autentikációhoz.

Hiba!

A rendelés megtekintéséhez kérjük, [jelentkezzen be](#).

Ételek megjelenése

A rendelés oldalon az elérhető ételek látványos és felhasználóbarát módon jelennek meg, hogy a felhasználók könnyen tudjanak választani a kínálatból. Minden ételhez tartozik egy étvágygerjesztő kép, amely vizuálisan segíti a döntést, valamint egy rövid leírás, amely ismerteti az étel főbb összetevőit vagy különleges jellemzőit. Az ár egyértelműen feltüntetésre kerül forintban, így a felhasználó pontosan láthatja a költségeket. Az oldalon gombok is találhatók, a „Hozzáadás a kosárhoz” gomb, amelynek segítségével a kiválasztott étel egyetlen kattintással hozzáadható a rendeléshez.



Marhapörkölt galuskával

Hagyományos magyar fogás, szaftos marhapörkölt puha galuskával.

Ár: 2500 Ft

Mennyiség:

- 1 +

Hozzáadás a kosárhoz

Mennyiség kezelése

A rendelés során a felhasználók nemcsak kiválaszthatják a kívánt ételleket, hanem beállíthatják azok mennyiségét is, ami különösen hasznos, ha egy adott fogásból több adagot szeretnének rendelni. Az oldalon elhelyezett plusz (+) és mínusz (-) gombok lehetővé teszik a mennyiség egyszerű növelését vagy csökkentését.

Hozzáadás a kosárhoz

A „Hozzáadás a kosárhoz” gomb megnyomása után a rendszer egy megerősítő üzenetet jelenít meg, amely rákérdez, hogy biztosan hozzá szeretné adni ezt az ételt a kosárhoz. Ez a lépés segít elkerülni a véletlen kattintásokat, és biztosítja, hogy a felhasználó tudatosan adja hozzá az adott tételt a rendeléséhez. A megerősítő ablakban a felhasználó eldöntheti, hogy folytatja-e a műveletet vagy mégsem kívánja kosárba helyezni az ételt.

A(z) localhost közlendője

Biztosan hozzá szeretné adni ezt a terméket a kosárhoz?

OK Mégse

Kosár

A kosár oldal célja, hogy átlátható és vizuálisan rendezett formában összesítse a felhasználó által kiválasztott ételeket. Itt a felhasználók részletesen megtekinthetik a kosárhoz hozzáadott ételeket, azok mennyiségét, árát, valamint a rendelés teljes összegét. Az oldal lehetőséget biztosít a rendeltételek mennyiségének módosítására vagy törlésére is, így a felhasználók még a véglegesítés előtt könnyedén alakíthatják a rendelésüket.

Tételek megjelenítése

A kosár oldal részletes, táblázatos formában jeleníti meg az összes kosárba helyezett ételt, feltüntetve az étel nevét, a rendelt mennyiséget, valamint a tételek együttes árát.

| Étel neve | Ár (Ft) | Mennyiség | Összeg (Ft) | Műveletek |
|-------------------------|---------|-----------|-------------|-----------|
| Marhapörkölt galuskával | 2500 | - 1 + | 2500 | Törlés |
| Pörkölt | 3000 | - 1 + | 3000 | Törlés |
| Teljes összeg: | | | 5500 Ft | |

Tételek eltávolítása, mennyiség változtatása

Lehetőség van az egyes ételek eltávolítására is: a „Törlés” gombra kattintva a kiválasztott tétel azonnal kikerül a kosárból, a felhasználók ezen az oldalon kényelmesen módosíthatják a hozzáadott ételek mennyiségét is. Ezen két funkció bármelyik használata után a végösszeg újraszámításra kerül

Tovább lépés a rendelési folyamatban

A kosár oldalán elhelyezett „Rendelés befejezése” gomb lehetővé teszi a felhasználók számára, hogy a kiválasztott ételek ellenőrzése után tovább lépjenek az összegzés oldalra. Ezen az oldalon történik meg a valódi rendelés véglegesítése, ahol a felhasználók megadják a szállításhoz szükséges adatokat, valamint kiválaszthatják a fizetési módot, végül megerősíthetik a rendelésüket.

Rendelés folytatása

Rendelés befejezése

A „Rendelés folytatása” gomb lehetőséget biztosít a felhasználóknak arra, hogy visszalépjenek a rendelés oldalára, ha még további ételeket szeretnének hozzáadni a kosárhoz. Ez a funkció különösen hasznos abban az esetben, ha a felhasználó meggondolja magát, vagy elfelejtett egy tételt kiválasztani, hiszen anélkül egészítheti ki rendelését, hogy újra kellene kezdenie a folyamatot.

A rendelési adatok összegzése

Az összegzés oldal célja, hogy részletes áttekintést nyújtson a rendelés minden eleméről, mielőtt a felhasználó végleges döntést hozna. Itt jelennek meg az ételnevek, azok mennyiségei, valamint a teljes rendelési összeg. Emellett a felhasználó itt adhatja meg a szállítási címét és a telefonszámát, illetve kiválaszthatja a fizetési módot. Ez a lépés kulcsfontosságú, mivel lehetőséget biztosít az adatok utolsó ellenőrzésére és szükség esetén módosítására, így biztosítva a hibamentes rendelés leadását. Az összegzés főbb funkciói

A rendelés összegző oldalán a felhasználó áttekintheti a kosár teljes tartalmát, beleértve az egyes ételek nevét, a rendelt mennyiséget, valamint az ezek alapján kiszámolt végösszeget. Az oldal alján jól láthatóan megjelenik az összesített végösszeg is, amely az összes kiválasztott tétel teljes árát tartalmazza. Ez az átlátható megjelenítés segít abban, hogy a felhasználó megbizonyosodjon a rendelés helyességéről, mielőtt véglegesíti azt.

Rendelt ételek:

Marhapörkölt galuskával

1 db

Pörkölt

1 db

Végösszeg: 5500 Ft

Szállítási adatok

A rendelés véglegesítése előtt a felhasználónak meg kell adnia a szállítási adatokat. Ezen a felületen kell kitöltenie a szükséges mezőket, hogy a rendelést véglegesíteni lehessen: Település (Megye), Város, Utca, Házszám, valamint a kapcsolattartás érdekében egy érvényes Telefonszám.

Település (Megye)

Város

Utca

Házszám

Telefonszám

Fizetési mód kiválasztása

Az összegzés oldalon a felhasználók kiválaszthatják a számukra legmegfelelőbb fizetési módot. Jelenleg a készpénzes fizetés az alapértelmezett és elérhető opció, amelyet a rendszer automatikusan kijelöl. A bankkártyás fizetés jelen állapotában pedig, még fejlesztés alatt áll.

Fizetési mód

Készpénz

Készpénz

Bankkártya (Fejlesztés alatt)

Utolsó lehetőség a módosításra

A felhasználónak az összegzés oldalon még egy utolsó lehetősége nyílik arra, hogy módosítsa a kosár tartalmát, amennyiben szeretne még változtatni a rendelésen. Ebben az esetben a „Rendelés módosítása” gomb segítségével visszaléphet a kosár oldalra, és hozzáadhat vagy eltávolíthat tételt.

Rendelés módosítása

Rendelés véglegesítése

Ha azonban úgy érzi a felhasználó, hogy minden adat helyes és a rendelés teljes, akkor a „Rendelés véglegesítése” gombra kattintva ténylegesen leadhatja a rendelést, amely ezután feldolgozásra kerül és átirányítja a felhasználót a számla oldalára.

Visszajelzés a felhasználónak

Sikeres adatbevitel esetén a rendszer automatikusan átirányítja a felhasználót a számla oldalra, ahol megtekintheti rendelése részleteit és visszaigazolást kap annak feldolgozásáról. Amennyiben azonban bármelyik kötelező mező kitöltése elmarad, az oldal figyelmeztető üzenetet jelenít meg: „Kérjük, töltsse ki ezt a mezőt!”, így biztosítva, hogy a rendelés csak hiánytalan adatok megadása után kerüljön véglegesítésre.

Település (Megye)

Város

!

Kérjük, töltsse ki ezt a mezőt.

A számla oldal főbb funkciói

A számla oldal egyfajta rendelési visszaigazolásként szolgál, amely megerősíti a felhasználó számára, hogy a rendelését a rendszer sikeresen feldolgozta. Az oldalon egy rövid üzenet jelenik meg, amely kifejezetten jelzi, hogy a rendelés rögzítésre került, és hamarosan kiszállításra kerül. Ezen kívül a felhasználó itt áttekintheti a rendelés legfontosabb adatait: a megrendelt ételek listáját, azok mennyiségét, az összesített végösszeget, valamint a megadott szállítási adatokat. Ez az oldal zárja le a rendelési folyamatot, biztosítva a felhasználó számára a teljes átláthatóságot és visszajelzést.

Számla megjelenítése

A **számla oldalon** a rendszer részletesen megjeleníti a rendelés adatait, hogy a felhasználó egyértelmű visszajelzést kapjon a megrendeléséről.

Az oldalon látható:

A felhasználó neve, valamint a megadott szállítási cím és telefonszám. Az összes megrendelt étel, azok mennyisége, valamint egységára. Az összesített végösszeg, amely az összes tétel alapján számított teljes fizetendő összeget mutatja. A rendelés dátuma és pontos időpontja, így a felhasználó visszakeresheti később, mikor adta le a rendelést.

Mátyás Csárda számla

Teljes Név: Nagy Ervin

Telefonszám: +30 111 1111

Kiszállítási cím: Barany, Pécs, Maléter Pál Utca 11

Fizetési mód: Készpénz

Rendelés dátuma: 2025-04-15 06:53:49

Rendelt Ételek:

- Marhapörkölt galuskával - 1 db, 2500 Ft
- Csirkepaprikás galuskával - 1 db, 2000 Ft

Végösszeg: 4500 Ft

Vissza az oldalra

A „Vissza az oldalra” gomb megnyomása után a kosár tartalma törlődik a felhasználó automatikusan átirányításra kerül a kezdőlapra, ahol a navigáció segítségével akár újra böngészheti az étlapot, megtekintheti profilját, vagy leadhat újabb rendelést. Ez a funkció segíti a gördülékeny navigációt a rendelés befejezése után.

Fejlesztői dokumentáció

Alkalmazott fejlesztői eszközök

HTML (HyperText Markup Language)

Szerepe:

A HTML (HyperText Markup Language) a webfejlesztés alapvető leírónyelve, amely a weboldal felépítését és tartalmi elemeit határozza meg.

Felhasználási területek:

A HTML a weboldal felépítésének és tartalmi elemeinek leírására használt leírónyelv, amely kulcsfontosságú szerepet tölt be az admin bejelentkezés és a felhasználói regisztráció felületeinek kialakításában. A regisztrációs és az admin bejelentkezés oldal HTML segítségével készülnek, melyek tartalmazzák a szükséges beviteli mezőket, például a felhasználónév, e-mail cím, jelszó megadására szolgáló mezőket, valamint a műveleteket elindító gombokat.

CSS (Cascading Style Sheets)

Szerepe:

A CSS biztosítja az oldal megjelenését, beleértve a színek, betűtípusok, távolságok és elrendezések szabályozását, így esztétikailag egységes és felhasználóbarát felületet hoz létre. Kiemelt szerepe van abban, hogy a különböző képernyőméreteken – legyen az mobil, tablet vagy asztali számítógép – a weboldal reszponzív módon jelenjen meg, vagyis alkalmazkodjon az adott eszközhöz. Ezen túlmenően a CSS lehetővé teszi, hogy az interaktív elemek, mint például a gombok vagy linkek különböző állapotai (pl. rámutatás, kattintás) vizuálisan visszajelzést adjanak a felhasználónak, ezzel is növelve a felhasználói élményt.

Felhasználási területek:

Az oldalak reszponzivitásának biztosításában, lehetővé téve, hogy azok különböző eszközökön – mint például mobiltelefonok, táblagépek vagy asztali számítógépek – megfelelően jelenjenek meg. Továbbá, a CSS lehetőséget ad interaktív elemek, például hover-effektusok alkalmazására, amelyek a felhasználói élmény javítása érdekében vizuális visszajelzéseket adnak.

JavaScript

Szerepe:

Az oldal dinamikus elemeinek működéséért való felelősség, mivel lehetővé teszi az interaktív funkciókat és az azonnali adatkezelést anélkül, hogy a weboldal újratöltésére lenne szükség. A JavaScript segítségével a felhasználói élmény zökkenőmentesebbé válik, mivel a felhasználó nem várakozik az oldal újratöltésére.

Felhasználási területek:

Felhasználási területei közé tartozik az adatok ellenőrzése az űrlapok beküldése előtt, például a regisztrációs vagy bejelentkezési űrlapoknál, biztosítva, hogy a szükséges mezők ki legyenek töltve. Api link megjelenítése a kezdő oldalon. Ezen kívül lehetővé teszi a dinamikus tartalom frissítését, mint például a kosárban lévő tételek mennyiségének módosítása, anélkül, hogy az oldal újra betöltődne. Végül a JavaScript alkalmazása interaktív elemek, például gombok működésére is, például a rendelési folyamatban, amelyek biztosítják, hogy a felhasználó könnyedén navigálhasson és végezhesse el a kívánt műveleteket az oldalon.

PHP (Hypertext Preprocessor)

Szerepe

A PHP a webalkalmazások backend logikájának megvalósítására szolgál, és a szerveroldali programozási nyelv egyik alapvető eszköze, amely az adatbázis-kezelésért és a dinamikus tartalom generálásáért felelős. A PHP segítségével az alkalmazás képes interaktívan kezelni a felhasználók adatait és biztosítani a weboldal megfelelő működését az adatbázissal való folyamatos kommunikációval.

Felhasználási területek:

Felhasználási területek közé tartozik a felhasználók bejelentkezési és regisztrációs adatainak kezelése, ami lehetővé teszi a felhasználók autentikálását és a személyre szabott élmények biztosítását. Emellett a PHP felelős az adatok lekérdezéséért és megjelenítéséért az adatbázisból, például éttermek menüinek, rendeléseinek vagy profiladatainak kezelésével. Az adatbázissal való kommunikáció során a PHP kéréseket küld és válaszokat fogad, így biztosítva a webalkalmazás dinamikus frissítését és az adatok valós idejű megjelenítését a felhasználók számára.

Példa php-n belüli SQL lekérdezésre, kommunikációra

```
// Az ételek lekérdezése az adatbázisból
$sql = "SELECT id, nev, leiras, ar, kep FROM etelek";
$result = $conn->query(query: $sql);
```

Fejlesztői környezet

Visual Studio Code (VS Code)

A Visual Studio Code (VS Code) egy könnyen használható, bővíthető szövegszerkesztő, amely a kódfejlesztés alapját képezi. A különböző kiegészítők és bővítmények segítségével jelentősen javíthatjuk a fejlesztési folyamat hatékonyságát. Ez a modern kódolási környezet nemcsak gyors, hanem rendkívül rugalmas is, lehetővé téve a felhasználók számára, hogy saját igényeikhez igazítsák a szerkesztőt.

Előnyök:

Az VS Code könnyedén integrálható a GitHub verziókezelő rendszerével, ami lehetővé teszi a kódok hatékony kezelését és együttműködést más fejlesztőkkel. Gyors hibakeresés és kódformázás. Az VS Code könnyedén integrálható a GitHub verziókezelő rendszerével, ami lehetővé teszi a kódok hatékony kezelését és együttműködést más fejlesztőkkel.

MySQL

A MySQL egy nyílt forráskódú adatbázis-kezelő rendszer, amely kulcsszerepet játszik az alkalmazás adatkezelésében. Az alkalmazás számára biztosítja az adatkezelés alapját, lehetővé téve az adatok hatékony és biztonságos tárolását és kezelését.

Példa a felhasználási területekre:

Felhasználók adatainak tárolása: Az alkalmazásban regisztrált felhasználók személyes adatai, például email címek és jelszavak tárolása történik az adatbázisban. A jelszavakat titkosítva tárolja, hogy biztosítsa a felhasználók biztonságát.

Ételek információinak tárolása: Az étterem étlapján található ételek részletes információi is a MySQL adatbázisban kerülnek tárolásra. Ilyen adatok például az étel neve, ára és rövid leírása, amelyek segítenek a felhasználóknak a választásban.

Verziókezelés és kommunikáció

Github

Szerepe:

A Github verziókezelési platformot használtuk a kódok megosztására, tárolására és nyomon követésére. Ez biztosította, hogy a csapattagok egyszerre dolgozhassanak a projekten. Verziókezelés (commitok és pull requestek kezelése), illetve csapatmunka elősegítése különböző ágak (branch-ek) használatával.

Discord

Szerepe:

A Discord csapatkommunikációs platformként szolgált, amely lehetőséget nyújtott a napi szintű kommunikációra és a fejlesztési folyamatok egyeztetésére.

Tervezési módszerek

A projekt tervezési szakasza elengedhetetlen volt az alkalmazás hatékony és sikeres megvalósításához. Ezek a tervezési módszerek biztosították, hogy az alkalmazás ne csak funkcionálisan, hanem vizuálisan is megfeleljen az elvárásoknak.

Felhasználói élmény (UX) és interfész (UI) tervezés

Felhasználói élmény (UX) tervezés

Cél

A tervezés során arra törekedtünk, hogy az alkalmazás használata intuitív és kényelmes legyen, minden felhasználói csoport számára. Az étlap, a rendelés és a profil oldalak logikai felépítését úgy alakítottuk ki, hogy a felhasználók gyorsan eligazodjanak és céljaikat minimális lépéssel elérjék.

Módszer

A felhasználói igények felmérése alapján készült egy tervezési vázlat. A navigáció logikája a legegyszerűbb struktúrára épült. Külön figyelmet fordítottunk arra, hogy a funkciók reszponzívak legyenek, tehát mobiltelefonokon, tableteken és számítógépeken egyaránt jól működjenek.

Interfész (UI) tervezés

Cél

Esztétikus, modern megjelenés létrehozása, amely illeszkedik az étterem hangulatához és karakteréhez. Az egyszerű színek, képek és vizuális elemek harmóniája a felhasználói élmény része.

Módszer

Színharmonia: A bordó (#8b0000) és világos bézs (#fdf5e6) színek dominálnak, amelyek eleganciát teremtenek. Betűtípusok: Az Arial-t és sans-serif betűtípusokat használtuk, amelyek könnyen olvashatók. Interaktív elemek: Gombok, hover-effektusok és más animációk kerültek beépítésre, amelyek fokozzák a vizuális élményt.

Drótvázak és vizuális prototípusok létrehozása

Drótváz készítés

Cél: Az oldalak szerkezetének megtervezése és az egyes funkciók elhelyezésének megtervezése.

Módszer

Vázlatok készítése az oldalstruktúra átláthatóságának érdekében. Az oldalakon a fő funkciók kiemelt helyre kerültek a fejlécben. A drótvázak alapján a logikai kapcsolatok (linkek, gombok) is meghatározásra kerültek. A wix.com eszközt használtuk a vizuális modellezéshez. Az egyes elemek elhelyezése és színekombinációk tesztelése során több verzió készült, amelyek közül a legjobban működőt választottuk ki.

Navigációs logika tervezése

Oldalak közötti navigáció

A navigációs logika célja, hogy biztosítsa a felhasználók számára a gyors, kényelmes és zökkenőmentes átjárhatóságot az alkalmazás különböző oldalai között. A jól megtervezett navigációs rendszer lehetővé teszi a felhasználók számára, hogy könnyedén és gyorsan elérjék a kívánt funkciókat és információkat, miközben minimalizálja a felesleges lépéseket.

Adatmodell és funkcionalitás tervezés

Az adatmodell és a funkcionalitás tervezésének célja a webalkalmazás adatainak hatékony kezelése és a felhasználói élmény optimalizálása. Az adatmodell kialakítása során figyelembe vettük, hogy az adatbázis logikai felépítése megfelelően tükrözze a felhasználói interakciókat és a rendszert működtető folyamatokat.

Az adatmodell tervezése során három fő táblát hoztunk létre: felhasználók, ételek és rendelések. Ezek a táblák közvetlenül kapcsolódnak egymáshoz, hogy biztosítsák az adatok közötti összhangot. A rendszer az adatokat 3. normál formába (3NF) szervezi, hogy minimalizáljuk az adatismétléseket és biztosítsuk a hatékony tárolást és lekérdezést. Minden tábla tartalmazza a szükséges attribútumokat, például a felhasználók táblája az e-mail címeket és jelszavakat, az ételek táblája az étel nevét, árát és leírását, míg a rendelések táblája tárolja az ételt, rendelési dátumot, valamint a rendeléshez tartozó mennyiséget.

A funkcionalitás tervezése az alkalmazás alapvető felhasználói folyamatainak logikai sorrendjét követi. A felhasználók számára az első lépés a regisztráció, amelyet a bejelentkezés követ. Ezt követően a felhasználók böngészhetnek az étlapot, majd hozzáadhatják az ételt a kosárhoz. A rendelési folyamat végén a kosárban lévő tételek véglegesítésére kerül sor, majd az összes rendelési adat megjelenítésre kerül a számla oldalon. A rendelési folyamat minden lépése biztosítja, hogy a felhasználó folyamatosan lássa a választott ételek összegét, és visszajelzést kapjon az aktuális állapotról.

Az adminisztrációs funkciók, mint az étlap szerkesztése és a rendelési előzmények megtekintése, elkülönítve vannak a felhasználói oldaltól, hogy biztosítsák az adminisztrátorok számára a megfelelő kontrollt és a rendszer működtetéséhez szükséges eszközöket.

Reszponzív dizájn tervezés

A részponzív dizájn célja, hogy biztosítsa az alkalmazás megfelelő megjelenését és működését különböző eszközökön, beleértve a mobiltelefonokat, táblagépeket és asztali számítógépeket. Az oldal részponzivitásának megvalósítására a Grid rendszerek, és a bootstrap alkalmazása volt a legjobb módszer, mivel ezek rugalmas elrendezéseket biztosítanak, amelyek alkalmazkodnak a különböző képernyőméretekhez.

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
```

Az alábbi kódban a CSS beállítások biztosítják a részponzivitást.

```
body {  
  font-family: 'Arial', sans-serif;  
  background: url('kepek/bejelentkezes.jpg') no-repeat center center fixed;  
  background-size: cover;  
  color: #4b3621;  
  height: 100vh;  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: center;  
  margin: 0;  
}
```

A Grid segítségével a komplexebb, több oszlopos elrendezések is megvalósíthatók, amely a részponzív dizájn során szükséges pontosságot és rendezettséget biztosít. A képernyő szélességétől függően az oldalelemek (mint például a gombok, képek, szövegek) automatikusan igazodnak, hogy a felhasználói élmény ne csorbuljon semmilyen eszközön. Pontosítva a részponzivitás a Bootstrap keretrendszer és az egyedi CSS kombinációjával valósul meg. A Bootstrap gondoskodik a részponzív viselkedésről, míg az egyedi CSS beállítások 'finomhangolják' az elrendezést és megjelenítést.

Összegzés

A tervezési módszerek biztosították, hogy az alkalmazás minden szempontból optimális működést és vizuális élményt nyújtson a felhasználók számára. Az UX (felhasználói élmény) és UI (felhasználói felület) tervezés segített abban, hogy a funkciók könnyen hozzáférhetők és intuitívan használhatók legyenek, míg az adatmodell és a funkcionalitás tervezése biztosította a rendszer technikai hátterét és a felhasználói adatkezelést. Az alkalmazás részponzív dizájnja lehetővé teszi a zökkenőmentes élményt minden eszközön, miközben a felhasználók számára minden szükséges információ és funkció könnyen elérhetővé válik.

Adatmodell leírása

Az adatbázis célja, hogy biztosítsa a felhasználók, adminok, rendelések és ételek kezelését egy hatékony és logikus struktúrában. A normalizálási szabályok figyelembevételével az adatbázis úgy lett megtervezve, hogy minimalizálja az adatismétlést és biztosítja az adatkonzisztenciát. Ennek érdekében a különböző adatokat egymással összefüggő táblákban tároljuk.

Adattáblák és struktúrájuk

adminok tábla

Az adminok tábla célja, hogy tárolja az alkalmazás adminisztrátorainak adatait. Az admin bejelentkezési oldala biztonsági okokból csak külön linken érhető el: http://localhost/Vizsga/admin_bejelentkezés.htm

Oszlopok és azok leírása

id: (Típus: INT, PRIMARY KEY): Ez az oszlop tartalmazza az adminisztrátor egyedi azonosítóját, amely automatikusan növekszik (auto increment), és biztosítja, hogy minden adminisztrátornak egyedi azonosítója legyen. Az érték nem lehet NULL.

adminnev: (Típus: VARCHAR(255)): Az adminisztrátor neve kerül ebbe az oszlopba. A mező nem lehet NULL, és minden adminisztrátornak kötelező egy nevet rendelni.

jelszo: (Típus: VARCHAR(255)): Ez az oszlop tárolja az adminisztrátor titkosított jelszavát. A jelszó biztonságos tárolása érdekében nem kerül tárolásra tiszta szöveggént, hanem titkosítva. Nem lehet NULL.

email: (Típus: VARCHAR(100)): Az adminisztrátor e-mail címe. Ez az érték egyedi kell hogy legyen, és nem lehet NULL, mivel az email címek biztosítják a kommunikációt és a belépést az adminisztrációs felületre.

| Oszlop neve | Típus | Leírás | Követelmények |
|-------------|-------------------|---|--------------------------|
| id | INT (PRIMARY KEY) | Egyedi azonosító, minden adminisztrátorhoz. | Auto Increment, nem NULL |
| adminnev | VARCHAR(255) | Adminisztrátor neve. | Nem NULL |
| jelszo | VARCHAR(255) | Titkosított jelszó. | Nem NULL |
| email | VARCHAR(100) | Admin e-mail címe. | Egyedi érték, nem NULL |

ételek tábla

Az ételek tábla tárolja az étlapon szereplő adatokat, és a rendelés során a rendeles.php oldal is innen kéri le a szükséges adatokat.

Oszlopok és azok leírása

id: (Típus: INT, PRIMARY KEY): Ez az oszlop az étel egyedi azonosítóját tartalmazza, amely automatikusan növekszik (auto increment). Az érték nem lehet NULL, biztosítva, hogy minden ételhez egyedi azonosító tartozzon.

nev (Típus: VARCHAR(255)): Az étel neve kerül ebbe az oszlopba. Mivel az étel neve alapvető információ, nem lehet NULL.

leiras: (Típus: TEXT): Az étel rövid leírását tartalmazza, amely bemutatja az ételt és annak összetevőit. Az oszlop NULL értéket elfogad, tehát nem kötelező minden ételhez leírást rendelni.

kep: (Típus: VARCHAR(255)): Ez az oszlop tárolja az étel képének elérési útját. A kép segíti a felhasználókat a választásban. Az oszlop NULL értéket elfogad, tehát nem minden ételnek szükséges képpel rendelkeznie.

ar: (Típus: INT): Az étel ára kerül ebbe az oszlopba forintban. Az ár mező nem lehet NULL, mivel ez alapvető információ az étel vásárlásához.

elkeszitheto: (Típus: TINYINT): Ez az oszlop jelzi, hogy az étel elérhető-e vagy sem. Az értéke 1 (TRUE), ha az étel elérhető, és 0 (FALSE), ha nem. Az adminisztrátor határozza meg, hogy egy étel elérhető-e, és az alapértelmezett érték 1 (elérhető).

Ez a tábla biztosítja az étlap kezelését, lehetővé téve az éttermek számára, hogy könnyedén hozzáadhassák, módosíthassák és eltávolíthassák az ételeket az étlapról.

| Oszlop neve | Típus | Leírás | Követelmények |
|--------------|-------------------|---|--------------------------|
| id | INT (PRIMARY KEY) | Egyedi azonosító minden ételhez. | Auto Increment, nem NULL |
| nev | VARCHAR(255) | Az étel neve. | Nem NULL |
| leiras | TEXT | Az étel rövid leírása. | Nem NULL |
| kep | VARCHAR(255) | Az étel képének elérési útja. | Nem NULL |
| ar | INT | Az étel ára. | Nem NULL |
| elkeszitheto | TINYINT | Az étel elérhetősége (admin határozza meg). | Default: TRUE(1) |

felhasznalok tábla

A felhasználók tábla tárolja a felhasználók személyes és bejelentkezési adatait, beleértve a titkosított jelszót is.

Oszlopok és azok leírása:

id (Típus: INT, PRIMARY KEY): Az egyedi azonosító minden felhasználóhoz. Az érték automatikusan növekszik (auto increment), és nem lehet NULL, biztosítva, hogy minden felhasználóhoz egyedi azonosító tartozik.

felhasznalonev: (Típus: VARCHAR(255)): A felhasználó egyedi felhasználóneve, amelyet regisztrációkor ad meg. Ez az érték egyedi, tehát nem lehet több felhasználónak ugyanaz a felhasználóneve, és nem lehet NULL.

jelszo: (Típus: VARCHAR(255)): A felhasználó titkosított jelszava, amely a hash segítségével van tárolva. Az érték nem lehet NULL, mivel a felhasználó bejelentkezéséhez elengedhetetlen.

email: (Típus: VARCHAR(255)): A felhasználó e-mail címe, amely szintén egyedi és nem lehet NULL. Az e-mail cím szükséges a felhasználó azonosításához, valamint a kommunikációhoz.

teljesnev: (Típus: VARCHAR(100)): A felhasználó teljes neve. Ez az oszlop nem lehet NULL, mivel az alkalmazásban szükséges a felhasználó pontos azonosítása.

Ez a tábla biztosítja a felhasználói adatok biztonságos tárolását, lehetővé téve a regisztráció, bejelentkezés és a felhasználói profil kezelését. A jelszavak titkosítása és a személyes adatok tárolása megfelelő biztonsági intézkedésekkel történik.

| Oszlop neve | Típus | Leírás | Követelmények |
|----------------|-------------------|---|--------------------------|
| id | INT (PRIMARY KEY) | Egyedi azonosító minden felhasználóhoz. | Auto Increment, nem NULL |
| felhasznalonev | VARCHAR(255) | A felhasználó egyedi felhasználóneve. | Egyedi érték, nem NULL |
| jelszo | VARCHAR(255) | Titkosított jelszó. | Nem NULL |
| email | VARCHAR(255) | A felhasználó e-mail címe. | Egyedi érték, nem NULL |
| teljesnev | VARCHAR(100) | A felhasználó teljes neve. | Nem NULL |

kosar tábla

A kosar tábla a felhasználók aktuális kosarának tartalmát tárolja, vagyis azokat az ételeket, amelyeket a felhasználó a rendelés véglegesítése előtt a kosárba helyezett. Minden kosárhoz tartozó tétel külön bejegyzésként szerepel, és a számlázás befejeztével az adott felhasználó kosarának adatai törlésre kerülnek.

Oszlopok és leírásuk

id: (Típus: INT, PRIMARY KEY): Egyedi azonosító minden kosártételhez. Az érték automatikusan generálódik (Auto Increment), és nem lehet NULL, ez biztosítja az egyes bejegyzések egyértelmű azonosítását.

felhasznalo_id: (Típus: INT): Ez az oszlop kapcsolatot jelent a **felhasznalok** táblával, megmutatva, hogy az adott kosártétel melyik felhasználóhoz tartozik. Ez egy idegen kulcs (Foreign Key), amely nem lehet NULL.

etel_id: (Típus: INT): Ez az oszlop kapcsolódik az **etelek** táblához, megadva, hogy melyik étel szerepel a kosárban. Szintén idegen kulcs (Foreign Key), és nem lehet NULL.

mennyiseg: (Típus: INT): Megadja, hogy az adott ételből hány darabot tett a felhasználó a kosárba. Ez az érték kötelező (nem NULL), mivel minden kosártételhez tartoznia kell egy meghatározott mennyiségnek.

Ez a tábla kulcsfontosságú a rendelési folyamat során, mivel innen történik a kosár tartalmának megjelenítése, módosítása, valamint az összesítés és véglegesítés előtt a rendelés összeállítása.

| Oszlop neve | Típus | Leírás | Követelmények |
|----------------|-------------------|--|--------------------------|
| id | INT (PRIMARY KEY) | Egyedi azonosító minden kosártételhez. | Auto Increment, nem NULL |
| felhasznalo_id | INT | Kapcsolat a felhasznalok táblával. | Nem NULL, Foreign Key |
| etel_id | INT | Kapcsolat az etelek táblával. | Nem NULL, Foreign Key |
| mennyiseg | INT | Az adott étel mennyisége a kosárban. | Nem NULL |

rendelesek tábla

A rendelések tábla azokat az adatokat tárolja, amelyek a felhasználók által leadott rendelésekhez kapcsolódnak. Minden egyesétel külön sorban jelenik meg, ezáltal lehetővé téve, hogy egy rendelésen belül többféle étel is szerepeljen külön mennyiséggel.

Oszlopok és részletes leírásuk:

id: (Típus: INT, PRIMARY KEY): Ez az oszlop biztosítja minden rendelés egyedi azonosítóját. Az érték automatikusan generálódik (AUTO_INCREMENT), és nem lehet NULL.

felhasznalo_id: (Típus: INT): Az adott rendelést leadó felhasználóhoz kapcsolódik. Idegen kulcs (FOREIGN KEY), amely a felhasznalok tábla id mezőjére hivatkozik. Nem lehet NULL.

etel_id: (Típus: INT): Az adott rendelésben szereplő étel azonosítója. Ez is egy idegen kulcs, amely az etelek tábla id mezőjéhez kapcsolódik. Szintén NOT NULL.

mennyiseg: (Típus: INT): Az adott étel megrendelt mennyiségét tárolja. Ez kötelező mező (NOT NULL), mivel minden rendeléshez meg kell adni a mennyiséget.

statusz: (Típus: VARCHAR(50)): A rendelés aktuális állapotát írja le. Alapértelmezett értéke „függőben”, de később módosítható például „Elfogadva” vagy „Elutasítva” értékekre.

rendeles_datuma: (Típus: DATETIME): A rendelés leadásának időpontját tartalmazza. Alapértelmezés szerint az aktuális időbélyeget (CURRENT_TIMESTAMP) rögzíti, és nem lehet NULL.

| Oszlop neve | Típus | Leírás | Követelmények |
|-----------------|-------------------|---|-----------------------------|
| id | INT (PRIMARY KEY) | Egyedi azonosító minden rendeléshez. | Auto Increment, nem NULL |
| felhasznalo_id | INT | Kapcsolat a felhasznalok táblával. | Nem NULL, Foreign Key |
| etel_id | INT | Kapcsolat az etelek táblával. | NOT NULL, Foreign Key |
| mennyiseg | INT | Az adott étel mennyisége a rendelésben. | NOT NULL |
| statusz | VARCHAR(50) | Rendelés státusza („Elfogadva”, „Elutasítva”). | Default: „függőben” |
| rendeles_datuma | DATETIME | A rendelés leadásának időpontja. | NOT NULL, CURRENT_TIMESTAMP |

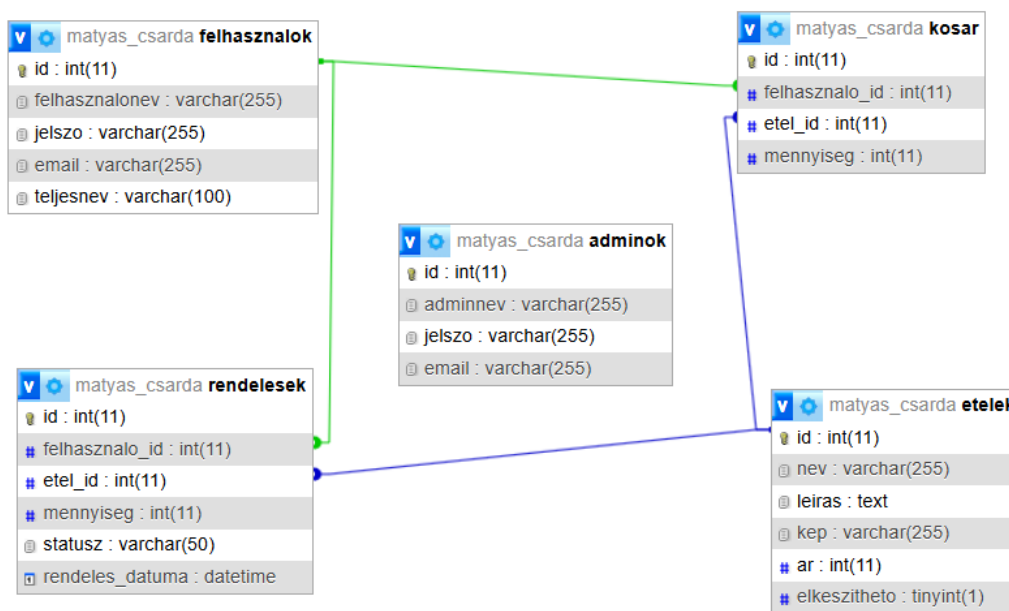
Kapcsolat az adattáblák között

Az adatbázisban a felhasználók és a kosár táblák között egy *egy-a-többhöz* ($1:N$) kapcsolat áll fenn, ami azt jelenti, hogy egyetlen felhasználó több különböző tételt is elhelyezhet a kosarában. Ez a kapcsolat technikailag a `felhasznalok.id` és a `kosar.felhasznalo_id` mezők között valósul meg.

A kosár és az ételek között szintén egy *egy-a-többhöz* ($1:N$) kapcsolat van, mivel egy adott étel több felhasználó kosarában is szerepelhet egyszerre. A kapcsolat a `etelek.id` és a `kosar.etel_id` mezők között jön létre.

A felhasználók és a rendelések táblái között ugyancsak egy *egy-a-többhöz* ($1:N$) reláció figyelhető meg, amely lehetővé teszi, hogy egy felhasználó több rendeltetést is leadjon. A kapcsolatot itt a `felhasznalok.id` és a `rendelesek.felhasznalo_id` mezők biztosítják.

Hasonló módon az ételek és a rendelések között is fennáll egy *egy-a-többhöz* ($1:N$) viszony, hiszen egyetlen étel többször is szerepelhet különböző rendelésekben. Ez a kapcsolat a `etelek.id` és a `rendelesek.etel_id` mezők között jön létre.



Részletes feladat-specifikáció és algoritmusok

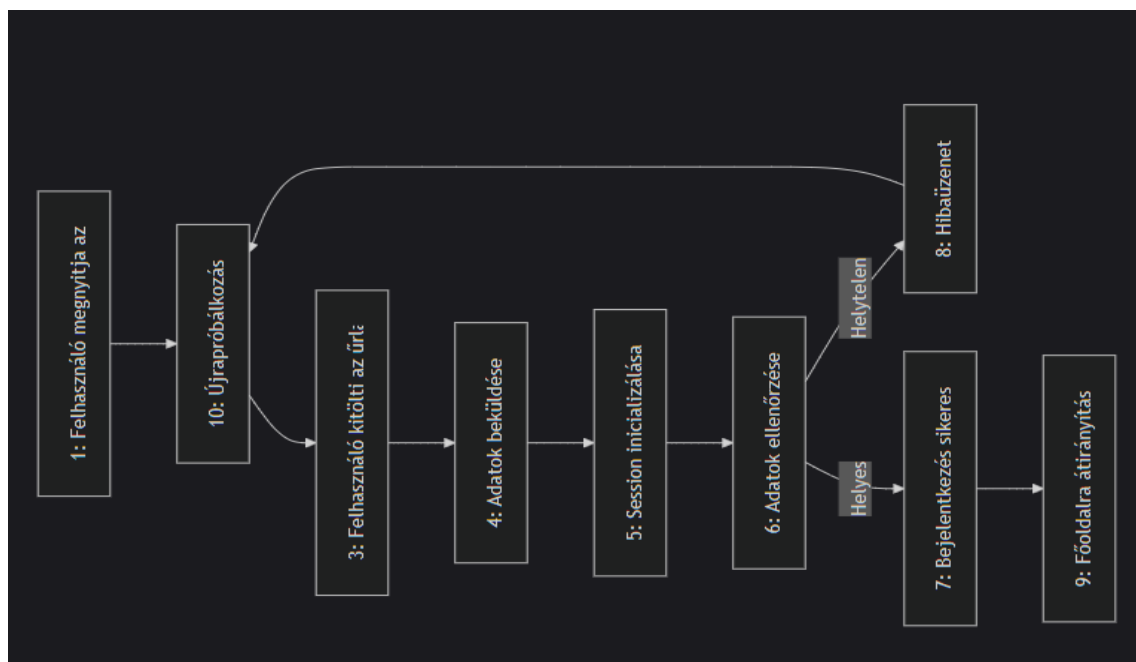
A részletes feladat-specifikáció és az algoritmusok célja, hogy az alkalmazás logikai működését és funkcióit pontosan meghatározzuk, valamint világos, átlátható formában dokumentáljuk. Ez a dokumentáció lehetővé teszi a fejlesztők számára, hogy könnyedén megértsék az alkalmazás működését, és ezáltal biztosítja a rendszer egyszerű továbbfejlesztésének lehetőségét a jövőben.

Részletes feladat-specifikáció

A regisztráció és bejelentkezés során a felhasználók egyedi felhasználónevet, jelszót és e-mail címet adnak meg, amelyek ellenőrzés után biztonságosan, titkosított formában kerülnek az adatbázisba. A bejelentkezést követően a felhasználók elérhetik személyes profiljukat és használhatják az alkalmazás funkcióit. A kosár funkció lehetővé teszi az ételek hozzáadását az aktuális rendeléshez, a mennyiségek módosítását, valamint a kosár tartalmának törlését vagy frissítését. A rendelési folyamat végén a felhasználó megadja a szállítási címét és a kívánt fizetési módot, majd a rendelés leadása után visszaigazolást kap számla formájában.

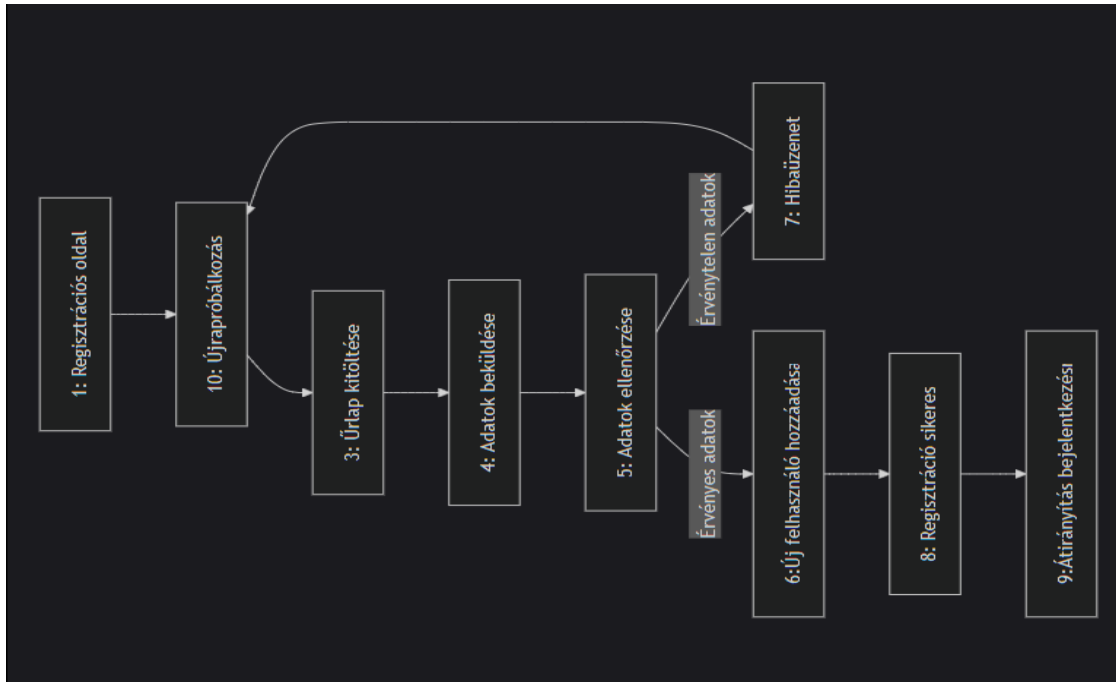
Bejelentkezés

A felhasználó megadja a felhasználónevet és jelszavát. A rendszer az adatbázisban ellenőrzi az adott felhasználóhoz tartozó jelszó hash-t, és összeveti a megadott jelszóval. Sikeres egyezés esetén a rendszer munkamenetet (session) indít, és belépteti a felhasználót. Sikertelen próbálkozás esetén hibaüzenet jelenik meg: „Hiba felhasználónév vagy jelszó”. Bejelentkezés után a felhasználó átirányításra kerül a főoldalra vagy a rendelési oldalra.



Regisztráció:

A felhasználó megadja a következő adatokat: egyedi felhasználónév, jelszó, jelszó megerősítése, e-mail cím, teljes név. A rendszer ellenőrzi, hogy a felhasználónév és e-mail cím nem szerepel-e már az adatbázisban. A jelszót a rendszer hash algoritmussal titkosítja, mielőtt adatbázisba kerülne. Sikeres regisztráció után visszajelzést kap a felhasználó, majd átirányítás történik a bejelentkezési oldalra.



Kosár

A felhasználó a kiválasztott ételekhez a „Hozzáadás a kosárhoz” gombra kattintva tud elemeket hozzáadni. Minden tételhez kapcsolódik egy mennyiség, amely alapértelmezetten 1, de a felhasználó módosíthatja azt. A rendszer minden kosártételhez elmenti a felhasználó azonosítóját (kapcsolat a felhasználók táblával) és az adott étel azonosítóját (kapcsolat az ételek táblával), így biztosított a személyre szabott tartalom. Ha a felhasználó ismét hozzáadja ugyanazt az ételt, a rendszer növeli a mennyiséget, nem hoz létre új sort. A kosár tartalma bármikor megtekinthető egy dedikált „Kosár” oldalon, ahol az alábbi műveletek lehetségesek:

Mennyiség módosítása: A felhasználó plusz vagy mínusz gombokkal módosíthatja az adott étel mennyiségét. **Tétel törlése:** Egyes elemek külön is eltávolíthatók. **Kosár ürítése:** Egyetlen gombbal az összes kosárelem törölhető. A kosár oldal automatikusan kiszámítja és megjeleníti az aktuális végösszeget. A kosár csak bejelentkezett felhasználók számára érhető el. Ha egy nem hitelesített látogató próbál hozzáadni tételt, a rendszer automatikusan átirányítja a bejelentkezési oldalra, és figyelmeztetést jelenít meg.

Rendelés folyamata

Kosár tartalmának átvizsgálása

Miután a felhasználó elvégezte a böngészést és hozzáadta a kívánt ételeket a kosárhoz, a rendelés folyamatának első lépése a kosár oldalon történik. Itt a felhasználó még egyszer ellenőrizheti a kiválasztott ételek mennyiségét és az összesített árat. A felhasználónak van lehetősége módosítani a mennyiségeken, eltávolítani tételt, vagy törölni az egész kosarat. Ha megfelelőnek találja a kosár tartalmát, tovább léphet az összegzés oldalára.

Szállítási és fizetési adatok megadása:

A felhasználó ezt követően a rendelés véglegesítéséhez szükséges adatokat adja meg: Az alkalmazás rögzíti a felhasználó által megadott települést, várost, utcát, házszámot és telefonszámot és ezeket a paramétereket átadja a számlázás oldalának amennyiben a felhasználó véglegesíti a rendelését. A felhasználónak két lehetősége van a fizetési mód kiválasztására: készpénzes fizetés (alapértelmezett) vagy bankkártyás fizetés (ha a jövőben implementálva lesz). A rendszer biztosítja, hogy minden adat érvényes legyen, és ha valamelyik mező üres marad, a felhasználó figyelmeztetést kap.

Rendelés összegzése:

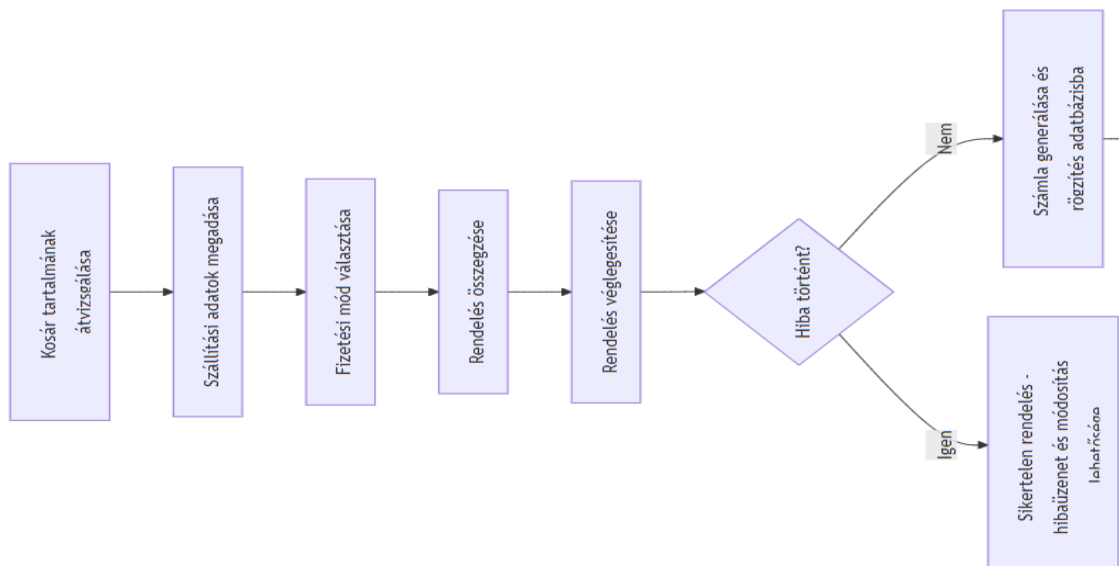
Az oldal automatikusan összegzi a rendelés teljes költségét, amely tartalmazza az ételek árát, és más költségeket, vagyis újra ellenőrizheti a rendelés részleteit, illetve még egyszer utoljára lehetősége van a felhasználónak, hogy módosítsa a rendelését.

Rendelés véglegesítése:

Miután a felhasználó minden adatot megadott, és meggyőződött a rendelés részleteiről, a „Rendelés véglegesítése” gomb megnyomásával a rendelés befejeződik. Ezzel egy időben átkerül a számlázási oldalra és a rendelés adatai átkerülnek az adatbázisba (rendelesek tábla), és a rendszer rögzíti a rendelés idejét, státuszát (kezdetben "függőben"), valamint a felhasználó adatokat és étkezési választásokat.

Számla generálása:

A rendelés véglegesítése után a felhasználó megkapja a rendelés visszaigazolását számla formájában, amely tartalmazza a megrendelt tételeket, azok árait, a szállítási és fizetési információkat. Ezen kívül minden rendeléshez hozzárendelésre kerül egy egyedi rendelési szám.



Adminisztrációs funkciók



Az adminisztrátorok számára külön felület biztosítja az étlap és a rendelések kezelését. Az étlap szerkesztésének keretében lehetőségük van új ételek hozzáadására, meglévő ételek eltávolítására, valamint az egyes tételek elérhetőségi státuszának (elérhető vagy elérhetetlen) beállítására. Ezen túlmenően az adminok nyomon követhetik a beérkezett rendeléseket, megtekinthetik azok részleteit és aktuális állapotát. A rendelési státuszokat is frissíthetik, például elfogadhatják vagy elutasíthatják azokat, biztosítva ezzel a rendelési folyamat átláthatóságát és gördülékeny lebonyolítását.

Mátyás Csárda Admin Panel

Kijelentkezés

Adminisztrációs Panel

Ételek Kezelése

| ID | Étel Neve | Leírás | Ár | Kép | Elérhetőség | Műveletek |
|----|-------------------------|---|---------|--|--------------|--|
| 1 | Pörkölt | Hagyományos sertéspörkölt galuskával. | 3000 Ft |  | Nem elérhető | <button>Módosítás</button> <button>Törlés</button> |
| 2 | Marhapörkölt galuskával | Hagyományos magyar fogás, szaftos marhapörkölt puha galuskával. | 2500 Ft |  | Elérhető | <button>Módosítás</button> <button>Törlés</button> |

Étel Szerkesztése

Étel neve:

Pörkölt

Leírás:

Hagyományos sertéspörkölt galuskával.

Ár:

3000

Kép neve:

porkolt.jpg

☐ Elérhetőség (pipáld ki, ha elérhető)

Mentés

Viszsa

Új Étel Hozzáadása

Beérkező Rendelések

| Rendelés ID | Felhasználó ID | Étel Neve | Mennyiség | Státusz | Műveletek |
|-------------|----------------|-------------------------|-----------|----------|--|
| 108 | 27 | Marhapörkölt galuskával | 1 | függőben | <div><div>Elfogadás</div><div>Elutasítás</div></div> |

Az adminnak nem szükséges látni több rendelési adatot mivel az már a futár dolga lenne.

Algoritmusok

A bejelentkezési és regisztrációs folyamat

Az első lépés az űrlapon bekért adatok (felhasználónév, jelszó, e-mail cím) érvényességének ellenőrzése. Regisztráció esetén az algoritmus megvizsgálja, hogy a megadott felhasználónév és e-mail cím már létezik-e az adatbázisban. Amennyiben nem, és a jelszavak is egyeznek a jelszót biztonságos formában, hash-eléssel tárolja el. Sikeres validálás után a felhasználó adatai bekerülnek az adatbázisba. Bejelentkezéskor az algoritmus lekéri a megadott felhasználónévhez tartozó hash-elt jelszót, majd összehasonlítja a felhasználó által beírt jelszóval. Sikeres egyezés esetén a rendszer munkamenetet (session) indít, és a felhasználó bejelentkezett állapotba kerül.

Kosár és rendelési folyamat algoritmus

A kosárkezelés algoritmus

A kosárkezelés algoritmus lehetővé teszi az ételek hozzáadását, mennyiségének módosítását, valamint azok eltávolítását a kosárból. Minden felhasználóhoz egyedi kosár tartozik, amely a session vagy az adatbázis alapján kerül lekérdezésre. Rendelés indításakor a rendszer ellenőrzi, hogy a kosár nem üres-e, majd az összegzés oldalon bekéri a szállítási adatokat és a fizetési módot. Az adatok beküldése után a rendszer összegzi a rendelés tartalmát, és lehetőséget biztosít az utolsó módosításokra. A véglegesítés során az algoritmus validálja az összes beviteli mezőt, majd rögzíti a rendelést az adatbázisban. Hiba esetén visszajelzést ad a felhasználónak, és lehetőséget nyújt az adatok javítására. Sikeres rendelés esetén a rendszer pedig számlát generál, és a rendelést „függőben” állapotban tárolja el az adatbázisban.

Adminisztrációs algoritmus

Az adminisztrációs modul algoritmus

Az adminisztrációs modul algoritmus lehetőséget biztosít az ételek adatainak dinamikus kezelésére. Az algoritmus az adminisztrátor hitelesítése után hozzáférést biztosít az étlap szerkesztéséhez, ahol új ételek felvitele során az admin megadhatja az étel nevét, árát, rövid leírását, valamint képet csatolhat a fájlrendszer vagy adatbázis segítségével. A meglévő ételek esetén az algoritmus lehetővé teszi a név, ár, leírás és státusz (elérhető/elérhetetlen) módosítását. Étél törlése esetén a rendszer eltávolítja azt az adatbázisból.

Tesztelési dokumentáció

A rendszer funkcionális tesztelése a PHPUnit tesztkeretrendszer segítségével történt, amely a PHP alkalmazások egység- és integrációs tesztelésének egyik legelterjedtebb eszköze. A tesztelés során különös figyelmet fordítottunk a legkritikusabb modulokra, mint például a bejelentkezés, regisztráció, kosárkezelés és rendelési folyamat.

Bejelentkezés

A PHPUnit tesztelés során a rendszer manuálisan feltöltötte az adatbázist előre definiált tesztadatokkal, amelyek segítségével a bejelentkezési folyamatot ellenőriztük. A teszt során a megadott felhasználói adatokkal történő belépés sikeresen, hibaüzenet nélkül lezajlott, ami azt igazolta, hogy a bejelentkezési logika helyesen működik és megfelelően kezeli az adatbázisban tárolt információkat.

```
Runtime:      PHP 8.2.12

.                                                     1 / 1 (100%)

Time: 00:00.159, Memory: 8.00 MB

OK (1 test, 2 assertions)
```

| | | id | felhasznalonev | jelszo | email | teljesnev |
|--------------------------|-----------|----|------------------|--|---------------------|-------------------|
| <input type="checkbox"/> | Módosítás | 25 | ErvinNagy | \$2y\$10\$Y5bssIZrevwJ3dBB050bJuaj7Hfiik80NUqKnBdvkT... | nagyervin@gmail.com | Nagy Ervin |
| <input type="checkbox"/> | Módosítás | 27 | GipszJakab | \$2y\$10\$8rEu44Q7maczkWFPbmNQaupFmnEX1kRnDQWEP2piF0... | gipszjakab@gmail | Gipsz Jakab |
| <input type="checkbox"/> | Módosítás | 30 | tesztfelhasznalo | \$2y\$10\$73X.13lcsYYaObTgywHDpuyuxee80Cq9z6S.oe1EDcY... | teszt@example.com | Teszt Felhasználó |

Regisztráció

A regisztrációs folyamat tesztelése során a PHPUnit teszt sikeresen végrehajtotta az új felhasználó regisztrációját az alkalmazásban. A megadott adatok nemcsak hibamentesen kerültek feldolgozásra, hanem a rendszer azokat megfelelően rögzítette is az adatbázisban, ezzel igazolva a regisztrációs logika helyes működését és az adatkezelés megbízhatóságát.

```
Runtime:      PHP 8.2.12

..                                                    2 / 2 (100%)

Time: 00:00.113, Memory: 8.00 MB

OK (2 tests, 2 assertions)
```

| | | id | felhasznalonev | jelszo | email | teljesnev |
|--------------------------|-----------|----|------------------------------|---|---------------------|-------------------|
| <input type="checkbox"/> | Módosítás | 25 | ErvinNagy | \$2y\$10\$Y5bssIZrevwJ3dBB050bJuaj7Hfiik80NUqKnBdvkT... | nagyervin@gmail.com | Nagy Ervin |
| <input type="checkbox"/> | Módosítás | 27 | GipszJakab | \$2y\$10\$8rEu44Q7maczkWFPbmNQaupFmnEX1kRnDQWEP2piF0... | gipszjakab@gmail | Gipsz Jakab |
| <input type="checkbox"/> | Módosítás | 56 | regisztráciotesztfelhasznalo | \$2y\$10\$zKUsc515Thg881Y.6AlmOsuUmlUauFEOuK2hYMgz8S... | teszt@example.com | Felhasználó Teszt |

Rendelés

A rendelési folyamat tesztelése során a PHPUnit teszt sikeresen azonosította, hogy regisztráció nélkül a felhasználó nem tud rendelést leadni, így az illetéktelen hozzáférés elleni védelem megfelelően működik

```
FOREIGN KEY (`felhasznalo_id`) REFERENCES `felhasznalok` (`id`))
```

A teszt ezt követően létrehozott egy új felhasználói fiókot, majd ismételt próbálkozva a rendelés leadásával a rendszer már hibamentesen fogadta el azt. A rendelési adatok sikeresen rögzítésre kerültek az adatbázisban, ami megerősíti a jogosultságkezelés és az adatfeldolgozás helyes működését.

```
Runtime:      PHP 8.2.12
.                                                     1 / 1 (100%)
Time: 00:00.044, Memory: 8.00 MB
OK (1 test, 6 assertions)
```

```
✓ PASSED: \RendelesTest::testSikeresRendeles
-----
Summary reported by PHPUnit => Tests: 1, Assertions: 6
Summary captured by extension => Tests: 1, Passed: 1
-----
```

| | id | felhasznalo_id | etel_id | mennyiseg | statusz | rendeles_datuma |
|---|-----|----------------|---------|-----------|----------|---------------------|
| <input type="checkbox"/> Módosítás Másolás Törlés | 106 | 1 | 1 | 2 | függőben | 2025-04-15 17:55:01 |

Admin bejelentkezés

Az adminisztrációs bejelentkezés tesztelése során a tesztprogram hibakód nélkül futott le, ami a bejelentkezési folyamat megfelelő működését igazolja. A tesztelés átláthatósága és egyszerűsége érdekében ebben az esetben nem alkalmaztam titkosított (hashed) jelszót, így közvetlenül ellenőrizhetővé vált a belépési logika működése. A teszt mind a sikeres, mind a sikertelen bejelentkezési próbálkozásokat lefedte, és ezek kimeneteit helyesen kezelte, ezáltal alátámasztva a rendszer biztonságos és pontos jogosultságellenőrzését.

```
Runtime:      PHP 8.2.12
..                                                    2 / 2 (100%)
Time: 00:00.050, Memory: 8.00 MB
OK (2 tests, 3 assertions)
C:\xampp\htdocs\Vizsga>
```

```
✓ PASSED: \AdminBejelentkezesTest::testHelyesBejelentkezes
✓ PASSED: \AdminBejelentkezesTest::testHibasBejelentkezes

-----
Summary reported by PHPUnit => Tests: 2, Assertions: 3
```

| | id | adminnev | jelszo | email |
|--------------------------|----|------------|-------------|------------------------|
| <input type="checkbox"/> | 1 | tesztadmin | tesztjelszo | tesztadmin@example.com |

Ételek kezelése az admin panelen belül

Az adminpanelen belüli ételkezelési funkciók tesztelése szintén sikeresen, 0 hibakóddal zárult, ami a funkcionalitás hibamentes működését igazolja. A tesztelés három fő részfolyamatra terjedt ki: az ételek törlésére, módosítására, valamint új ételek hozzáadására.

| | id | nev | leiras | kep | ar | elkeszitheto |
|--------------------------|----|------------|---------------------------------|-------------|------|--------------|
| <input type="checkbox"/> | 28 | Palinka | Saját készítésű szilva pálinka. | palinka.jpg | 600 | 1 |
| <input type="checkbox"/> | 42 | Teszt étel | Teszt leírás | teszt.jpg | 2000 | 1 |

Mindhárom művelet esetében a rendszer megfelelően hajtotta végre a kívánt műveletet, az adatbázisban is helyesen rögzültek a változások, így a tesztelés eredménye alapján az admin felület ételkezelési modulja stabilan és megbízhatóan működik.

```
Runtime:      PHP 8.2.12
...
Time: 00:00.094, Memory: 8.00 MB
OK (3 tests, 10 assertions)
```

```
✓ PASSED: \EtelekKezeleseTest::testEtelHozzaadasa
✓ PASSED: \EtelekKezeleseTest::testEtelModositasa
✓ PASSED: \EtelekKezeleseTest::testEtelTorlese

-----
Summary reported by PHPUnit => Tests: 3, Assertions: 10
```


Továbbfejlesztői lehetőségek

A jövőbeli továbbfejlesztési tervek célja, hogy a rendszer még korszerűbbé, felhasználóbarátabbá és üzleti szempontból is versenyképesebbé váljon. Az egyik legfontosabb lépés a bankkártyás fizetés integrálása, amely lehetővé tenné a felhasználók számára, hogy a megrendelt ételek összegét közvetlenül, online fizessék ki. Ez nemcsak kényelmesebbé, hanem biztonságosabbá is tenné a tranzakciókat, valamint hozzájárulna a rendelési folyamat gyorsabb lebonyolításához. Emellett fontos fejlesztési irány az ételek kategorizálása mind az étlapon, mind a rendelés felületén. A kategóriák – például levesek, főételek, desszertek, vegetáriánus vagy gluténmentes opciók – bevezetésével az étlap strukturáltabbá válna, így a felhasználók egyszerűbben és gyorsabban megtalálhatnák az igényeiknek megfelelő ételeket.

A számlázási rendszer jelenlegi állapotában csupán egy egyszerű számlát generál a rendelés véglegesítése után, amely nem tekinthető hivatalos számlának. A hivatalos elektronikus számlázás bevezetése fontos lépés lenne a rendszer professzionális szintre emelése érdekében, hiszen ez lehetővé tenné a megfelelő könyvelési adminisztrációt és a hatósági előírásoknak való megfelelést.

További jövőbeli lehetőségként szerepel egy automatikus asztalfoglalási rendszer kialakítása, amely lehetővé tenné a felhasználók számára, hogy előre lefoglalják helyüket az étteremben. Ez különösen hasznos lehet a nagyobb forgalmú időszakokban, például hétvégén vagy ünnepnapokon, amikor a vendégek biztosítani szeretnék az ülőhelyüket. Az asztalfoglalás akár a rendeléshez is kapcsolódhatna, így a rendszer egyszerre kezelné az ételrendelést és a vendégek fogadását, ezáltal hatékonyabbá téve az étterem működését és a kiszolgálást.

Források, segítség

A projekt során kiemelt figyelmet fordítottunk a szerzői jogok tiszteletben tartására és a plagizálás elkerülésére, ezért képet – kizárólag mesterséges intelligencia, azon belül is a Copilot segítségével generáltunk. A fejlesztés folyamán több kódrészlet elkészítésében is szerepet játszott a promptalapú támogatás, amely során mesterséges intelligenciával generált ötleteket használtunk kiindulási alapként. A PHP-kódolás során gyakran támaszkodtunk a [w3schools.com](https://www.w3schools.com) weboldalon található dokumentációkra és példákra, amelyek segítséget nyújtottak a szintaxis pontos alkalmazásához és bizonyos funkciók implementálásához. A JavaScript kódok megírásában pedig a [javascripttutorial.net](https://www.javascripttutorial.net) oldal kínálta útmutatók szolgáltak hasznos forrásként, amikor elakadtunk egy-egy megoldandó feladat során. A tesztelési szakaszban a PHPUnit tesztkeretrendszer alkalmaztuk, és annak hatékonyabb használata érdekében a fejlesztői környezetünkbe beépítettük a PHPUnit Test Workbench kiterjesztést is, amely nagyban megkönnyítette és gyorsabbá tette a tesztek futtatását. Mindezek a források és eszközök jelentősen hozzájárultak ahhoz, hogy a fejlesztés átlátható, jogilag tiszta és szakmailag megalapozott módon történjen. A térkép megjelenítéséhez szükséges api linket pedig az alábbi oldalról származik: <https://leafletjs.com/>.