

WEISS MANFRÉD

BKSZC Weiss Manfréd Technikum, Szakképző Iskola és
Kollégium

Vizsgaremek



Készítette: Szabó Zsuzsanna,
Tóth Viktor,
Patakfalvi Ádám

Témavezető: Kovács László

Tartalomjegyzék

Bevezetés.....	3
Témaválasztás indoklása	3
Célkitűzés	3
Kiknek szánjuk a weboldalt	3
Fejlesztői dokumentáció	4
Fejlesztői környezet	4
Domain és tárhely	5
Github és Git környezet	13
Kialakított adatszerkezet	14
Adatbázis táblái	14
A 'recept' tábla	15
A 'kep' tábla	16
A 'user' tábla	17
Algoritmusok a backend és frontend megvalósításban	18
Felhasználó regisztrációs folyamat backend oldal	18
Felhasználó regisztrációs folyamat frontend oldal	20
Recept feltöltési folyamat backend oldal	22
Recept feltöltési folyamat frontend oldal	24
Különböző körülmények, esetek és hibakezelések	26
Keresési folyamat kezelése	26
Belépési folyamat kezelése	27
Fejlesztési lehetőségek	27
Felhasználói dokumentáció	29
Üdvözlök!	29
Szükséges eszközök a Recept24 weboldal használatához	30
Weboldal eléréséhez használható böngészők	30
Weboldal elérhetősége	30
Weboldal használatának részletes ismertetése	31

Recept keresés folyamatának ismertetése	32
Regisztráció folyamatának ismertetése	34
Belépés folyamatának ismertetése	36
Recept feltöltés folyamatának ismertetése	38
Összefoglalás	40
Elkészült munkánk értékelése	40
Köszönetnyilvánítás	41
Irodalomjegyzék	42

Témaválasztás indoklása

A receptes weboldal ötlete a csapat szenvedélyes főzős-sütni iránti szeretetén alapszik. Az egyik csapattag, aki nagyon szeret főzni és sütni, felvetette az ötletet, hogy mi lenne, ha létrehoznánk egy olyan online platformot, ahol recepteket oszthatunk meg és inspirálhatjuk egymást.

Célkitűzés

Célunk egy könnyen kezelhető receptes weboldal létrehozása, amely lehetővé teszi a felhasználók számára a receptek keresését, megosztását és véleményezését. A felhasználóknak lehetőségük van saját receptek feltöltésére, beleértve a recept leírását és képeket is. Azonban a regisztráció mellett további előnyök is várják a felhasználókat, például egyedi profil létrehozása, ami lehetővé teszi számukra a belépést saját profiljukba, hogy könnyen hozzáférjenek a receptekhez és véleményezésekhez. A regisztrált felhasználók képesek lesznek véleményezni a recepteket, megosztani tapasztalataikat és javaslatokat tenni, így aktív és interaktív közösség alakulhat ki az oldalon.

Kiknek szánjuk a weboldalt

A weboldalunkat minden olyan főzős-sütni szeretőnek szánjuk, akik szeretnének tanulni, bővíteni receptkönyvüket, vagy egyszerűen csak élvezni a gasztronómia világát.

Fejlesztői környezet

A receptes weboldalunk elkészítéséhez különböző fejlesztőeszközöket használunk, amelyek segítenek a fejlesztésben és az adatok kezelésében. A Visual Studio Code egy könnyen használható kódszerkesztő, amely segít a weboldal kialakításában.

A XAMPP egy olyan program, amely tartalmazza az Apache webkiszolgálót, a MySQL adatbáziskezelőt és a PHP-t, és lehetővé teszi számunkra, hogy lokálisan fejlesszük és teszteljük weboldalunkat a saját számítógépünkön anélkül, hogy szükség lenne valós webkiszolgálóra.

A phpMyAdmin segítségével pedig könnyedén kezelhetjük és karbantarthatjuk az adatbázisunkat.

Végül a MySQL adatbázis-kezelő rendszer hatékonyan tárolja és kezeli a receptekhez, felhasználókhöz és egyéb adatokhoz kapcsolódó információkat.

Ezeket a fejlesztőeszközöket választottuk, mert tanultunk velük, és ismerjük őket. A széles körű elérhetőségük és az, hogy ingyenesen elérhetők, tovább erősíti döntésünket. Segítségükkel hatékonyan fejleszthetünk és tesztelhetünk weboldalunkat, és könnyedén kezelhetjük az adatbázist.

Domain és tárhely

Az online elérhető honlapunk megvalósíthatósága miatt szükséges volt egy domain, és egy tárhely kezelőt keresni. Választásunk a Nethely-re (<https://www.nethely.hu/>) esett, mert mindkét igényünket ki tudta elégíteni. Az ingyenes tárhely szolgáltatása mellé egy ingyenes domaint is adva, teljesen lefedte a szükségleteinket. Így jutottunk el a fiók létrehozása <http://recept24.szakdoganet/> regisztrációjáig.

(A lenti folyamatok dokumentálása miatt teljesen előlről, egy recept24 regisztrációval lett végig vezetve.)

Az email címet és hozzá való jelszó kérés után, egy email cím ellenőrzés következett, amiben található ellenőrző link, átvetetett az alábbi egyszerűen és jól érthetően prezentált létrehozást segítő oldalra:

Üdvözljük a Nethely ügyfélszolgálatán

Indítsa el internetes megjelenését 3 egyszerű lépésben.

1) Fiók hitelesítés



2) Tárhely

Kapcsolja be az első tárhelyét.

INGYENES TÁRHELY

TÁRHELY CSOMAGOK

3) Domain

Végül válasszon domain nevet.

INGYENES DOMAIN

DOMAINKERESŐ

Ingyenes tárhely

A Nethelynél biztosítunk Önnek 1 db ingyenes tárhelyet, amely belépő szintű megoldást kínál a weboldalak üzemeltetésének világában.

✓ 256 MB tárhely

✓ dinamikus

✓ reklámmentes

BEKAPCSOLÁS



Fontos tudnivalók az ingyenes tárhelyről:

- A szolgáltatás bekapcsolásával Ön elfogadja az [Általános Szerződési Feltételeket](#)
- A szolgáltatás bekapcsolásával Ön elfogadja az [Adatvédelmi és adatkezelési szabályzatot](#)
- Nem jár le, bármennyig használhatja
- Teljesen reklámmentes
- Dinamikus, CMS kompatibilis
- Akár saját domain nevét is hozzárendelheti, sőt még az sem feltétel, hogy az a Nethelynél legyen regisztrálva
- Biztonsági mentést **nem** készítünk a tárolt adatokról

Tárhely létrehozás (bekapcsolás) után:

Üdvözljük a Nethely ügyfélszolgálatán

Indítsa el internetes megjelenését 3 egyszerű lépésben.

1) Fiók hitelesítés 	2) Tárhely 	3) Domain Végül válasszon domain nevet. <div>INGYENES DOMAIN</div> <div>DOMAINKERESŐ</div>
--	---	--

A felajánlott al domain-ek közül a feladatunknak megfelelő címrészt választottuk

Ingyenes domain

A Nethelynél biztosítunk Önnek 1 db ingyenes domaint, hogy szolgáltatásainkat teljeskörűen kipróbálhassa. Ezzel a címmel elérhetővé teheti feltöltött weboldalát, kipróbálhatja levelező rendszerünket és egyéb szolgáltatásainkat is.

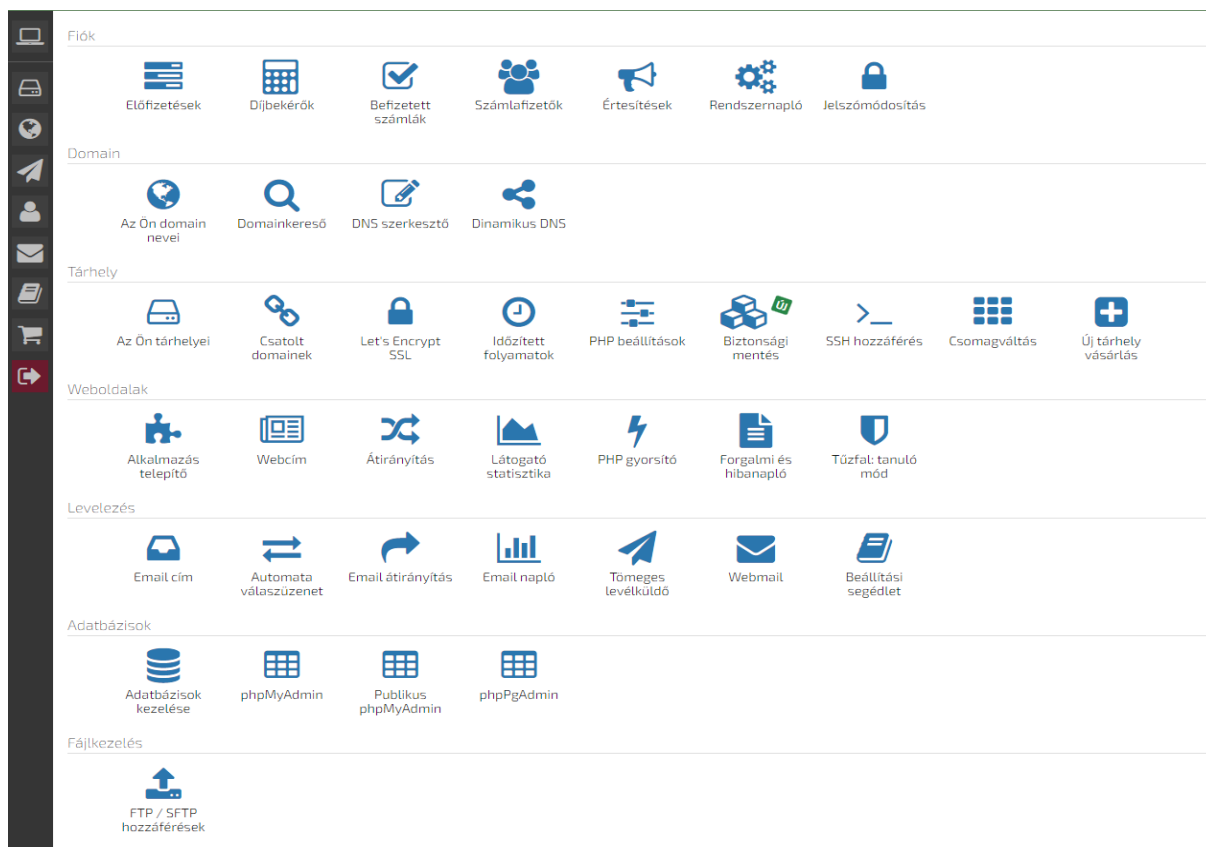
recept24	.szakdoga.net	LÉTREHOZÁS
----------	---------------	------------

Fontos tudnivalók az ingyenes domainről:

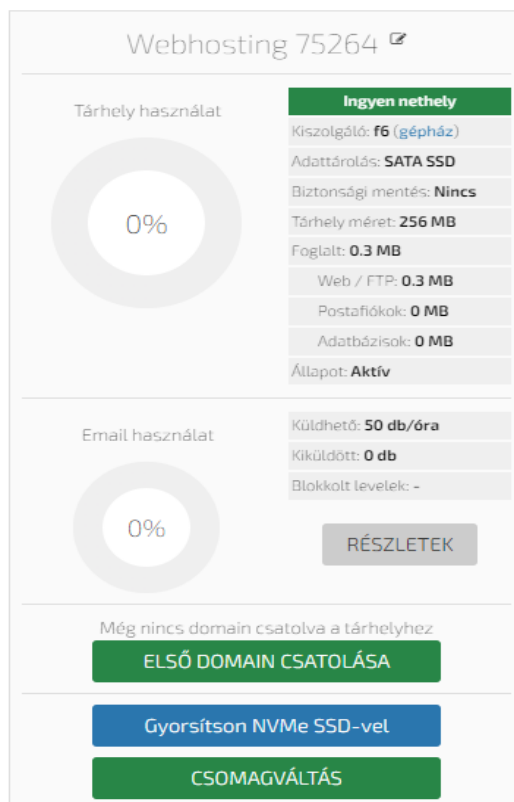
- A szolgáltatás létrehozásával Ön elfogadja az [Általános Szerződése Feltételeket](#)
- A szolgáltatás létrehozásával Ön elfogadja az [Adatvédelmi és adatkezelési szabályzatot](#)
- Nem jár le, bármeddig használhatja
- Tárhelyhez rendelve nem számoljuk bele az elhelyezhető domainek számába
- Az ingyenes domain teljes értékű domain, melyen weboldalak, email címek működhetnek
- A domain előtagját Ön határozza meg (ELŐTAG.nhely.hu, ELŐTAG.probaljaki.hu, ELŐTAG.webtelek.hu, ELŐTAG.szakdolgozat.net, ELŐTAG.szakdoga.net vagy ELŐTAG.infora.hu)

'Domain kezelés' műszaki beállítása elkészült.
(recept24.szakdoga.net)

A fenti folyamatok után a vezérlőpult ikon oldala jött fel, jellemzően a sötét színű ikonokat használtuk a lentebb leírt beállításaink során:



A tárhellyel kapcsolatos alapadatokat is megtekinthetjük.



Első Domain Csatolás-ra kattintva:

Tárhely / Csatolt domainek / Új

Domain csatolás

NethelyNethelynél kezelt domainek

- válasszon -

Csatolni kívánt domain

A legördülő lista már tartalmazza is, a korábban igényelt domaint:

Domain csatolás

NethelyNethelynél kezelt domainek

recepit24.szakdoga.net

- válasszon -

recepit24.szakdoga.net

recepit24.szakdoga.net

Mentve:

Információ

Az ingyenes domain nevek beállításait teljesen automatikusan kezeljük, Önnek nincs teendője.

BEZÁRÁS

'Domain kezelés' műszaki beállítása elkészült.
(recepit24.szakdoga.net)

Webcím beállítása, hogy a tárhely mely részében (mappájában) keresse a kezdő lapot, és egyéb beállítások adhatók meg:

Tárhely / Webcím / Új

Webcím

Előtag

Domain

recept24.szakdoga.net

Az előtag nélküli verzióhoz hagyja üresen az 'Előtag' mezőt és csak a domaint válassza ki.

SSL tanúsítvány

(új)

Nincs kiválasztható SSL tanúsítvány

Let's Encrypt SSL

Beállítások

PHP verzió

PHP 8.2

Karakterkódolás

Automatikus

Egyedi IP cím

Egyedi IP cím

Nincs kiválasztható IP cím

Könyvtár

A könyvtár kiválasztásával megadhatja, hogy tárhelyén belül hol szeretné elhelyezni weboldala forrását. A megadott könyvtár tartalma fog megjelenni a böngészőben a cím beírását követően.

Nincs alkönyvtár.

+ ÚJ KÖNYVTÁR

Naplózás

A funkció bekapcsolásával elérhetővé válnak az oldalról készült naplófájlok a 'Forgalmi és hibnapló' menüpontban.
FIGYELEM! A naplófájlok a tárhelyén kerülnek elhelyezésre, az általuk foglalt hely beleszámít a tárhelyhasználatba.
TIPP: Csak akkor válassza a 'Forgalom naplózása' vagy a 'Teljes naplózás' opciót, ha a forgalmi adatokra valóban szüksége van.

Naplózás

Nincs

Tűzfal

A tűzfal egy folyamatosan frissülő szabályrendszerrel szűri weboldala forgalmát, így védve meg a kártékony lekérdezésektől. A funkció bekapcsolásával biztonságosabbá teheti weboldalt.
FIGYELEM! A tűzfal esetében a weboldala normál funkcióit is blokkolhatja, ennek javításához az azonnali kikapcsolás helyett először próbálja meg konfigurálni a beállításokat a **tanuló mód** használatával.

☒ Tűzfal engedélyezése

Megjegyzés

MENTÉS

MEZŐK TÖRLÉSE

Mentés után pár pillanattal az alábbi értesítések jelentek meg:

'Átirányítás' műszaki beállítása elkészült.
(www.recept24.szakdoga.net ->
http://recept24.szakdoga.net)

'Webtárhely' műszaki beállítása elkészült.
(Azonosító: vh-75264, Kiszolgáló: f6, Megnevezés:
Webhosting 75264)

'Domain kezelés' műszaki beállítása elkészült.
(recept24.szakdoga.net)

'Aldomain' műszaki beállítása elkészült.
(recept24.szakdoga.net)

9

A webcím oldalon már megjelentek a beállítottak:

Tárhely / Webcím

< 1 >

Cím

http://recept24.szakdog.net

Könyvtár: /

Cím

Ezek után az adatbázis beállítása következett:

Tárhely / SQL Adatbázis / Új

Belépési adatok

Adatbázis név és felhasználó

Jelszó

Használjon erős jelszót

Típus és karakterkódolás

Adatbázis típus

MySQL

Karakterkódolás

A karakterkódolás a későbbiekben nem módosítható.

UTF-8

Megjegyzés

Távoli elérés

A Nethelynél létrehozott adatbázisait akár távolról is elérheti. Kapcsolódhat bármilyen programból, adatbázis kezelő szoftverből. Csak akkor nyissa ki adatbázisát, ha erre valóban szüksége van.

Az alábbi listába felveheti azokat az IP címeket, amelyekre korlátozni szeretné az adatbázis távoli hozzáférhetőségét. Ha szükséges megadhat *-ot is, ebben az esetben bárholnnan kapcsolódhat, bár ezt a megoldást csak indokolt esetben javasoljuk (pl. nincs fix IP címe).

Kapcsolódási adatok (MySQL)

Kiszolgáló: mysql.nethely.hu

Port: 3306

Kapcsolódási adatok (PostgreSQL)

Kiszolgáló: psq1.nethely.hu

Port: 5432

A távoli elérés funkció nem érhető el ezen a tárhelyen.

CSOMAGVÁLTÁS

Publikus phpMyAdmin

A funkció bekapcsolásával engedélyezheti az adatbázis kezelőbe történő belépést felhasználónév és jelszó megadásával. A publikus phpMyAdmin felület a <https://mysqladmin-pub.nethely.hu> címen érhető el.

FONTOS! A funkció bekapcsolásával engedélyezi, hogy a phpMyAdmin felületen keresztül bárholnnan elérhető legyen az adatbázisa. Csak akkor engedélyezze a funkciót, ha erre valóban szüksége van! A zárt phpMyAdmin verzió gombnyomásra elérhető minden adatbázishoz.

A publikus phpMyAdmin funkció nem érhető el ezen a tárhelyen.

CSOMAGVÁLTÁS

Beírva az adatokat, majd mentés után már létre is jött az adatbázis:

Tárhely / SQL Adatbázis							
Adatbázis	Felhasználó	Típus	Méret	Karakterkódolás	Távoli elérés 0	Publikus phpMyAdmin 0	Műveletek
recept24	recept24	MySQL	0 Számítás...	UTF-8	A funkció nem érhető el ezen a tárhelyen.	A funkció nem érhető el ezen a tárhelyen.	Beállítások
Adatbázis	Felhasználó	Típus	Méret	Karakterkódolás	Távoli elérés	Publikus phpMyAdmin	Műveletek

Oldal 1 / 1

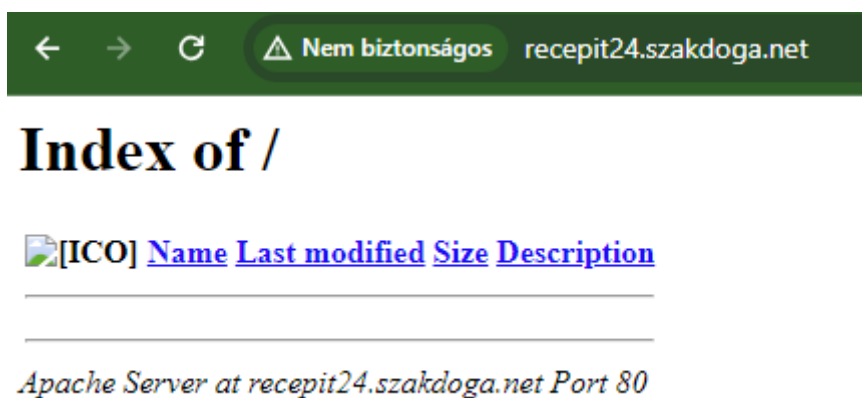
Kis idő, 1-2 percre rá, már be is lehetett lépni az adatbázisba a "phpMyAdmin" gombra kattintva:

Tárhely / SQL Adatbázis							+ ÚJ ADATBÁZIS	TUDÁSBÁZIS
<div>< 1 ></div>							10 sor / oldal	<input type="text"/>
Adatbázis	Felhasználó	Típus	Méret	Karakterkódolás	Távoli elérés	Publikus phpMyAdmin	Műveletek	
recept24	recept24	MySQL	Számítás...	UTF-8	A funkció nem érhető el ezen a tárhelyen.	A funkció nem érhető el ezen a tárhelyen.	phpMyAdmin	Beállítások
							CSOMAGVÁLTÁS	CSOMAGVÁLTÁS
Adatbázis	Felhasználó	Típus	Méret	Karakterkódolás	Távoli elérés	Publikus phpMyAdmin	Műveletek	

Oldal 1 / 1

A phpMyAdmin felületen már azonos a fentebb említett XAMPP segítségével elérhetővel.

Ezen teendők közben eltelt idő után már a domaint beírva a böngészőbe:



Már létezik a domain.

A fejlesztés megkönnyítése érdekében szükséges az FTP kapcsolat beállítása a Nethely kezelőfelületén, hogy pl.: Total Commander-rel kényelmesen másolhassuk át a fejlesztett fájljainkat.

Első lépésként itt is szükséges egy alap hozzáférés kialakítása:



Beállítva:

Tárhely / FTP, SFTP / Új

Belépési adatok

Felhasználónév
recept24

Jelszó
.....
35% - Gyenge

Protokoll
FTP/SFTP választása esetén az FTP felhasználóval mindkét protokoll használható.
FTP

Megjegyzés

Könyvtár

A könyvtár kiválasztásával megadhatja az FTP hozzáférés gyökérkönyvtárát a tárhelyén belül. A megadott könyvtárból a felhasználó nem tud kilépni, így biztosíthatja, hogy csak azon adatok legyenek elérhetők számára, amik a megadott könyvtárban belül vannak. A beállításnak többfelhasználós környezetben van igazán jelentősége.

Nincs alkönyvtár.


+ ÚJ KÖNYVTÁR

Mentve, megjelenik a kapcsolat:

Tárhely / FTP, SFTP						+ ÚJ FELHASZNÁLÓ	TUDÁSBÁZIS
<div>< 1 ></div>						10 sor / oldal	<input type="text"/>
Felhasználó	Belépés engedélyezve	Könyvtár	FTP kiszolgáló	Protokoll	Műveletek		
recept24	<input checked="" type="checkbox"/> Igen	/	ftp.nethely.hu	FTP	<input type="button" value="Beállítások"/> <input type="button" value="Töröl"/>		
Felhasználó	Belépés engedélyezve	Könyvtár	FTP kiszolgáló	Protokoll	Műveletek		

Oldal: 1 / 1

A jobb felső sarokban a helyfüggő tudásbázist választva:

 Kapcsolódó témák a tudásbázisban

FTP csatlakozás Total Commander segítségével
4.3
30718 megtekintés

Hálózati hely csatolása Windows 10-ben
FTP felhasználó felhasználásával
2.8
13760 megtekintés

FTP és SFTP felhasználó létrehozása az ügyfélszolgálatban
3.7
10883 megtekintés

SFTP csatlakozás WinSCP használatával
4.0
8461 megtekintés

Könnyen kiválasztható, hogy milyen beállításokkal szükségesek, pl.: a fentebb is említett Total Commander részére, hogy elérhessük könnyen távolról a tárhelyet.

Github és Git környezet

A kiírt igénynek eleget téve és közös munkánk zökkenőmentes végzése miatt, kialakítottunk egy GitHub környezetet a fejlesztésünknek.

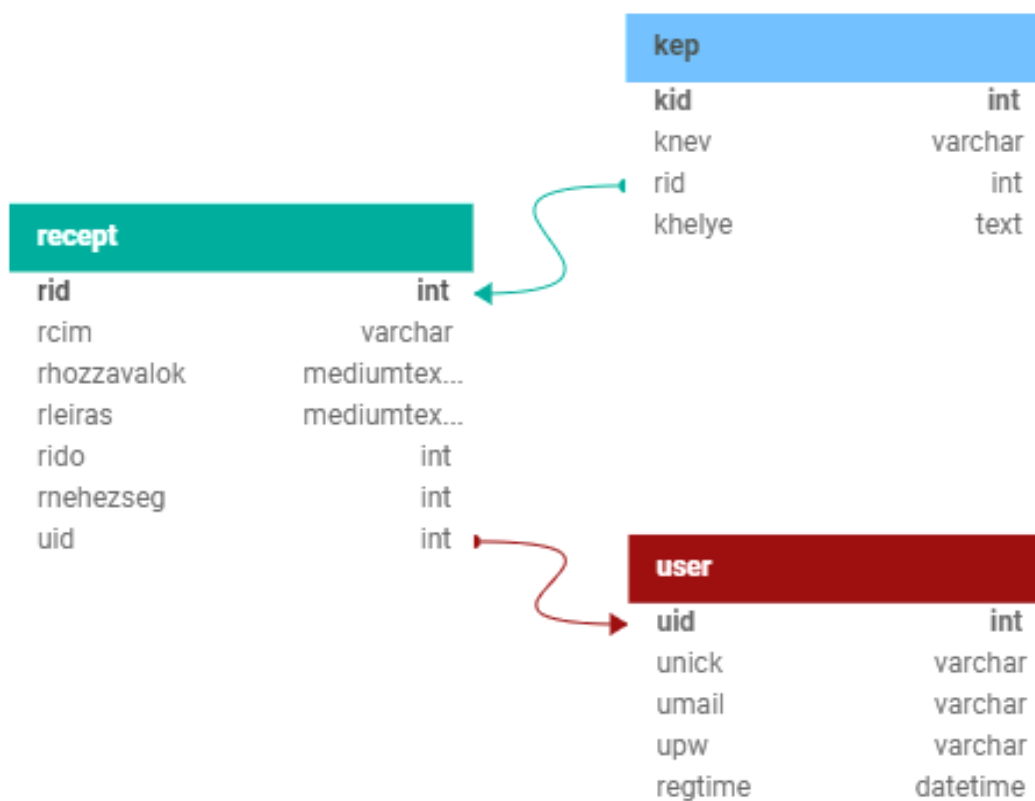
A környezetet egy közösen létrehozott email cím regisztrálásával végeztük el. Belépve egy külön repository-t hoztunk létre Recept24 néven. Ebben, mint résztvevőkként felvettük a saját accountjainkat, így mindenkinek a fejlesztés során végrehajtott változásai könnyen követhetőkké váltak. És a fejlesztés során létrejött egyéb fájlok közös elérhetősége is megoldhatóvá vált.

Egyéni környezeten, a <https://git-scm.com/> helyről telepített eszköz lehetőséget adott a saját gépünk és a felhős tárhely közötti Git szabályainak megfelelő kapcsolatot.

Kialakított adatszerkezet és részletes bemutatása

A receptes weboldalunk adatbázisa több táblát tartalmaz. Ezek a táblák strukturáltak és szervezettek, hogy hatékonyan kezeljék és tárolják az adatokat.

Adatbázis táblái



A 'recept' tábla

recept	
rid	int
rcim	varchar
rhozzavalok	mediumtex...
rleiras	mediumtex...
rido	int
rnehezseg	int
uid	int

rid (Recept azonosító): Ez a mező az adott recept egyedi azonosítója a táblában. Ez az elsődleges kulcs, amely automatikusan növekszik minden új recept létrehozásakor, és egyedi az adott recepthez. Az egyedi azonosító segít megkülönböztetni és azonosítani a különböző recepteket a táblában.

rcím (Recept címe): Ez a mező tárolja a recept címét, amely segít azonosítani és megkülönböztetni az egyes recepteket. A recept címe megjelenik a

receptek listázásakor, és segít a felhasználóknak megtalálni a kívánt receptet.

rhozzavalok (Recept hozzávalók): Ez a mező tárolja a recept hozzávalóit, általában egy hosszabb szöveges mező formájában. Ide tartoznak azok az összetevők és alapanyagok, amelyeket a recept elkészítéséhez szükséges.

rleiras (Recept leírása): Ez a mező tartalmazza a recept lépéseit vagy útmutatásait, amelyeket a felhasználóknak követniük kell a recept elkészítése során. Ez a rész adja meg azokat az utasításokat vagy lépéseket, amelyek segítségével a felhasználók létrehozhatják és elkészíthetik.

rido (Recept időtartama): Ez a mező tárolja a recept elkészítésének várható időtartamát, általában percekben vagy más időegységekben. Ez segít a felhasználóknak felmérni, hogy mennyi időre van szükségük a recept elkészítéséhez.

rnehezseg (Recept nehézségi szintje): Ez a mező jelzi a recept nehézségi szintjét vagy bonyolultságát. Általában egy szám típusú mező, amely tartományban vagy értékben jelzi a recept nehézségét, például könnyű, közepes vagy nehéz.

uid (Felhasználó azonosító): Ez a mező az idegen kulcs a táblában, ami összekapcsolja a recepteket a felhasználókkal. Az user_id alapján lehet kapcsolatot kialakítani a két tábla között, és lehet látni, hogy mely felhasználók hozták létre az egyes recepteket. Ez segít abban, hogy a felhasználók megnézhessék a saját létrehozott receptjeiket.

A 'kep' tábla

kep	
kid	int
knev	varchar
rid	int
khelye	text

kid (Kép azonosítója): Ez a mező az adott kép egyedi azonosítója a táblában. Az "kid" mező az elsődleges kulcs, ami azt jelenti, hogy minden képhez egyedi azonosítót rendel. Ez segít az egyes képek azonosításában és hivatkozásában más táblákban.

knev (Kép neve): Ez a mező a képek nevét vagy címét tárolja a rendszerben. A "knev" mező legfeljebb 40 karaktert fogad el, és a VARCHAR adattípust használja az adatbázisban, ami lehetővé teszi a változó hosszúságú szövegek tárolását. Ez segíti az egyedi képek azonosítását és megkülönböztetését, hiszen mindegyik képhez külön név vagy cím van rendelve. Így könnyen megtalálhatjuk az adott képet az adatbázisban, amikor szükség van rá.

rid (Recept azonosítója): Ez a mező azonosítja azt a receptet, amelyhez a kép tartozik. Az adott képhez egy recept van rendelve, és ez a mező tartalmazza a kapcsolódó recept azonosítóját. Ez lehetővé teszi a képek összekapcsolását a hozzájuk tartozó receptekkel.

khelye (Kép helye): Ez a mező tárolja a kép fizikai helyét vagy elérési útvonalát. A "khelye" mező TEXT típusú, ami azt jelenti, hogy hosszabb szöveges adatokat tud tárolni, például a kép elérési útvonalát. Ez a mező segít abban, hogy az alkalmazás megtalálja és megjelenítse a képet a weboldalon.

A 'user' tábla

user	
uid	int
unick	varchar
umail	varchar
upw	varchar
regtime	datetime

uid (Felhasználó azonosító): Ez a mező az adott felhasználó egyedi azonosítója a táblában. Az "uid" mező egy szám típusú, és az elsődleges kulcsa a táblának. Automatikusan növekszik minden új felhasználó regisztrációja során, és segíti az egyedi felhasználók azonosítását és megkülönböztetését.

unick (Felhasználónév): Ez a mező tárolja a felhasználó által választott felhasználónevet. A "unick" mező VARCHAR típusú, és legfeljebb 32 karakter hosszú lehet. Ezt a felhasználónevet használja a rendszer a felhasználók azonosítására és megkülönböztetésére.

umail (E-mail cím): Ez a mező tárolja a felhasználó által megadott e-mail címet. A "umail" mező VARCHAR típusú, és legfeljebb 80 karakter hosszú lehet. Az e-mail címek segítenek a felhasználók azonosításában és a kommunikációban velük.

upw (Jelszó): Ez a mező tárolja a felhasználó jelszavát. A "upw" mező VARCHAR típusú, és legfeljebb 40 karakter hosszú lehet. Fontos, hogy a jelszavakat biztonságosan tároljuk az adatbázisban, hogy minimalizáljuk a biztonsági kockázatokat.

regtime (Regisztráció ideje): Ez a mező rögzíti a felhasználó regisztrációjának idejét. A "regtime" mező DATETIME típusú, és az adott időpontot rögzíti minden új felhasználó regisztrációja során. Segít nyomon követni a felhasználók regisztrációs időpontját és tevékenységeit a rendszerben.

Algoritmusok a backend és frontend megvalósításban

Általában a dokumentációk főként a backend részre összpontosítanak, de az algoritmusok működése és azok interakciója a frontend és backend között is fontos szerepet játszik az alkalmazás teljes rendszerének megértésében. A frontend rész bemutatása segíthet a felhasználóknak és a fejlesztőknek is abban, hogy átfogó képet alkothassanak az alkalmazás működéséről az adott alkalmazás funkcióin keresztül.

Felhasználó regisztrációs folyamat backend oldal

```
<?php
// Create user hozzáadása az adatbázishoz:

header('Content-Type: application/json; charset=utf-8');

// fő logika, az eredmény kimenetekkel:
if (!check_fields()) {
    $createuser = ['hiba' => "Nincs kitöltve megfelelően a regisztráció"];
} elseif (!free_user()) {
    $createuser = ['hiba' => "Létezik már ilyen regisztráció!"];
} else {
    user_create();
    $createuser = ['hiba' => 0];
}

$json = json_encode($createuser, JSON_UNESCAPED_UNICODE);
print $json;

/*ellenőrzése, hogy minden mező ki van-e töltve. szuperglobális($_POST)
be van-e állítva és nem üres, azaz be van állítva valami?
*/
function check_fields()
{
    return isset($_POST["unick"]) && !empty($_POST["unick"]) &&
        isset($_POST["umail"]) && !empty($_POST["umail"]) &&
        isset($_POST["upw"]) && !empty($_POST["upw"]);
}

// szabad-e a felhasználónév, illetve emailcím.
function free_user()
{
    $unick = $_POST["unick"];
    $umail = $_POST["umail"];

    //adatbázis kapcsolat felépítése
    require '../connection/index.php';

    $search = mysqli_query($conn, "SELECT unick, umail FROM user WHERE
    unick = '$unick' OR umail = '$umail'");
    $row = mysqli_fetch_assoc($search);

    mysqli_close($conn);
```

```

    return !isset($row);
}

// felhasználó beírása az adatbázisba.
function user_create()
{
    /*a POST függvény kiolvasása, a htmlspecialchars() segítségével
    bizonyos karakterek ", ', <, > kikódolásával:
    */
    $unick = htmlspecialchars($_POST["unick"]);
    $umail = htmlspecialchars($_POST["umail"]);
    $upw = $_POST["upw"];

    //adatbázis kapcsolat felépítése
    require '../connection/index.php';

    //injektálás https://www.php.net/manual/en/mysqli-stmt.bind-param.php
    $stmt = "INSERT INTO user (unick, umail, upw, regtime)
    VALUES (?, ?, ?, NOW())";

    $stmt_prepare = mysqli_prepare($conn, $stmt);

    mysqli_stmt_bind_param($stmt_prepare, "sss", $unick, $umail, $upw);

    mysqli_stmt_execute($stmt_prepare);

    mysqli_close($conn);
}
?>

```

A header beállításoknál a Content-Type fejléc JSON formátumra történő beállítása történik. Az űrlap mezők ellenőrzésekor (check_fields) ellenőrizzük, hogy a POST metódussal érkező adatokban megtalálhatóak-e a kötelező mezők: felhasználónév (unick), e-mail (umail) és jelszó (upw). Ha bármelyik mező hiányzik vagy üres, a regisztráció nem folytatódik. A felhasználói adatok szabad már léteznek-e (free_user) lépésben ellenőrizzük, hogy a felhasználó által megadott felhasználónév vagy e-mail cím már létezik-e az adatbázisban más felhasználóhoz kötve. Ha már foglaltak, a regisztráció sikertelen lesz. A felhasználó létrehozásakor (user_create) az adatokat beillesztjük az adatbázisba, beleértve a felhasználónév, e-mail cím, jelszó és regisztráció időpontját. Végül JSON formátumban visszajelzést küldünk a regisztrációs folyamat eredményéről. Ha a regisztráció sikeres volt, a felhasználó adatait küldjük vissza, ellenkező esetben hibaüzenetet adunk vissza, például ha a felhasználónév vagy e-mail cím már foglalt volt. Ez a folyamat biztosítja a felhasználók regisztrációjának hatékony és biztonságos kezelését a rendszerben, amelyet a backend rész gondoskodik az adatok helyes kezeléséről és azok biztonságos rögzítéséről az adatbázisban.

Felhasználó regisztrációs folyamat frontend oldal

```
<?php

if(isset($_POST['upw']))
{
    $pwd = $_POST['upw'];
    $hashedPassword = md5($pwd, $binary = false);

    $apiUrl = 'http://recept24.szakdoganet/api/createuser/index.php';

    $userAgent = 'Mozilla/5.0 (Windows NT 11.0; Win64; x64) AppleWebKit/537.36
    (KHTML, like Gecko) Chrome/99.0.9999.99 Safari/537.36';

    $data = [
        'umail' => $_POST['umail'],
        'unick' => $_POST['unick'],
        'upw'   => $hashedPassword,
    ];

    $ch = curl_init($apiUrl);

    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($data));
    curl_setopt($ch, CURLOPT_USERAGENT, $userAgent);

    $response = curl_exec($ch);
    $responseData = json_decode($response);
    print_r($responseData);
    if ($response === false) {
        echo 'CURL request failed: ' . curl_error($ch);
    } else {
        if ($responseData->hiba == "0")
        {
            echo "<h2>Sikeres regisztráció!</h2>";
            echo '<a href="?p=login">Bejelentkezés!</a>';
        }
        else
        {
            print($responseData->hiba);

            //header("Refresh:0; url=http://recept24.szakdoganet/?p=register.php");
        }
    }
}

curl_close($ch);
}
```



```

if (!isset($_POST['upw'])) || (isset($_POST['upw']) && ($responseData->hiba != "0"))
{
    ?>
    <div class="container">
    <div class="form">
    <div class="title">Recept24</div>
    <div class="subtitle">Regisztráció!</div>

    <form method="post" action="?p=register" id="register">
    <div class="input-container ic1">
    <input type="email" id="umail" name="umail" class="input" required placeholder=" ">
    <div class="cut cutshort1"></div>
    <label class="placeholder" for="umail">Email</label>
    </div>
    <div class="input-container ic2">
    <input type="text" id="unick" name="unick" required class="input" placeholder=" ">
    <div class="cut"></div>
    <label class="placeholder" for="unick">Felhasználónév</label>
    </div>

    <div class="input-container ic2">
    <input type="password" id="upw" name="upw" required class="input" placeholder=" ">
    <div class="cut cutshort2"></div>
    <label class="placeholder" for="upw">Jelszó</label>
    </div>

    <button type="text" class="submit">Regisztrál</button>
    <div class="subtitle">Van már fiókja? <?php echo '<a href="?p=login">Bejelentkezés
    </a>' ?></a></div>
    </form>
    </div>
    </div>

    <?php
    echo '</div>';
}
?>

```

Az űrlap megjeleníti a regisztrációs űrlapot, amikor a felhasználó először megnyitja az oldalt vagy újratölti azt, és tartalmazza az e-mail cím, felhasználónév és jelszó mezőket, amelyeket a felhasználó kitölt.

Amikor a felhasználó elküldi az űrlapot a regisztrációs adatokkal, a frontend először hash-eli a jelszót biztonságos továbbítás érdekében, majd összeállítja az adatokat a backend számára, ideértve az e-mail címet, felhasználónevet és a hashelt jelszót.

A frontend POST kérést küld az adatokkal a backend API-jához a regisztráció végrehajtására, majd várakozik a válasza a backend-től az API kérésre.

Ha a válasz sikeres, azaz a regisztráció feldolgozása és rögzítése az adatbázisban sikeresen megtörtént, a frontend opcionálisan megjeleníti a sikeres regisztrációs üzenetet vagy az API

által visszaküldött adatokat. Ellenkező esetben, ha a válasz sikertelen, például a backend nem érhető el vagy hiba történt a regisztráció során, a frontend hibaüzenetet jelenít meg a felhasználónak. A frontend CURL kérést használ az adatok küldésére és a válasz fogadására a backend API-jához. Ha a CURL kérés sikertelen (például hálózati hiba vagy hibás URL), a frontend hibaüzenetet jelenít meg a felhasználónak. Ez a folyamat biztosítja, hogy a felhasználók regisztrációs adatai megfelelően és biztonságosan kerüljenek továbbításra és rögzítésre a backend rendszerben, és hogy a frontend felhasználóbarát módon kommunikáljon a felhasználóval a regisztrációs folyamat során fellépő eseményekről.

Recept feltöltési folyamat backend oldal

```
<?php
// Recept hozzáadása az adatbázishoz:

header('Content-Type: application/json; charset=utf-8');

// Fő logika, az eredmény kimenetekkel:
if (check_fields()) {
    // Ha minden mező ki van töltve, létrehozuk a receptet
    recipe_create();
    // Visszajelzés létrehozása az eredménnyel
    $createrecipe = [
        ['hiba' => 0],
        [
            'rcim' => $_POST["rcim"],
            'rhozzavalok' => $_POST["rhozzavalok"],
            'rleiras' => $_POST["rleiras"],
            'rido' => $_POST["rido"],
            'rnehezseg' => $_POST["rnehezseg"],
            'uid' => $_POST["uid"]
        ]
    ];
} else {
    // Ha valamelyik mező nincs kitöltve, hibát jelzünk
    $createrecipe = ['hiba' => "Nincs kitöltve megfelelően a recept"];
}

// Ellenőrzi, hogy minden mező ki van-e töltve
function check_fields()
{
    // Ellenőrzi, hogy minden mező tartalmaz-e értéket a $_POST tömbben
    $rcim = isset($_POST["rcim"]) && !empty($_POST["rcim"]);
    $rhozzavalok = isset($_POST["rhozzavalok"]) && !empty($_POST["rhozzavalok"]);
    $rleiras = isset($_POST["rleiras"]) && !empty($_POST["rleiras"]);
    $rido = isset($_POST["rido"]) && !empty($_POST["rido"]);
    $rnehezseg = isset($_POST["rnehezseg"]) && !empty($_POST["rnehezseg"]);
    $uid = isset($_POST["uid"]) && !empty($_POST["uid"]);

    // Visszatérési érték true, ha minden mező ki van töltve, különben false
    return $rcim && $rhozzavalok && $rleiras && $rido && $rnehezseg && $uid;
}
```

```

// Recept létrehozása az adatbázisban.
/* A POST adatok kiolvasása és kódolása, majd kapcsolat felépítése az adatbázissal.
*/
function recipe_create()
{
    /* A POST függvény kiolvasása, a htmlspecialchars() segítségével bizonyos
    | karakterek ", ', <, >  kikódolásával:
    */
    $rcim = htmlspecialchars($_POST["rcim"]);
    $rhozzavalok = htmlspecialchars($_POST["rhozzavalok"]);
    $rleiras = htmlspecialchars($_POST["rleiras"]);
    $rido = htmlspecialchars($_POST["rido"]);
    $rnehezseg = htmlspecialchars($_POST["rnehezseg"]);
    $suid = htmlspecialchars($_POST["uid"]);

    // Adatbázis kapcsolat felépítése
    require '../connection/index.php';

    // Beszúrási utasítás előkészítése
    $stmt = "INSERT INTO recept (rcim, rhozzavalok, rleiras, rido, rnehezseg, uid)
    VALUES (?, ?, ?, ?, ?, ?)";

    $stmt_prepare = mysqli_prepare($conn, $stmt);

    // Paraméterek összekapcsolása a beszúrási utasítással
    mysqli_stmt_bind_param($stmt_prepare, "sssiis", $rcim, $rhozzavalok, $rleiras,
    $rido, $rnehezseg, $suid);

    // Beszúrási végrehajtása
    mysqli_stmt_execute($stmt_prepare);

    // Adatbáziskapcsolat bezárása
    mysqli_close($conn);
}

// Válasz JSON formátumba alakítása és kiírása
$json = json_encode($createrecipe, JSON_UNESCAPED_UNICODE);
print $json;
?>

```

Az első lépésben a válasz Content-Type fejlécét application/json értékre állítja, ami jelzi, hogy a válasz JSON formátumban lesz visszaadva.

Ezután a fő logika ellenőrzi, hogy minden szükséges mező ki van-e töltve a POST kérésben. Ha minden mező ki van töltve, akkor létrehozza a receptet a recipe_create() függvény segítségével, majd összeállítja a választ a sikeres regisztráció adataival. Ha valamelyik mező nincs megfelelően kitöltve, akkor hibaüzenetet állít elő.

A check_fields() függvény biztosítja, hogy minden szükséges mező tartalmaz-e értéket a \$_POST tömbben, és visszatérési értéke true, ha minden mező ki van töltve, különben false.

A recipe_create() függvényben a POST adatokat kikódolja bizonyos karakterekkel a biztonságos adatbázisműveletek érdekében. Ezután felépít egy kapcsolatot az adatbázissal,

előkészíti a beszúrási utasítást a recept táblába, majd végrehajtja a beszúrást és bezárja az adatbáziskapcsolatot.

Végül az eredményt JSON formátumba alakítja és kiírja a választ a kliens felé. Ez a folyamat biztosítja, hogy a regisztráció során az adatokat biztonságosan rögzítse az adatbázisban, és a válasz JSON formátumban kerüljön visszaadásra a kliensnek.

Recept feltöltési folyamat frontend oldal

```
<?php

if (isset($_POST['rcim']))
{
    $data = [
        'rcim' => $_POST['rcim'],
        'rhozzavalok' => $_POST['rhozzavalok'],
        'rleiras' => $_POST['rleiras'],
        'rido' => $_POST['rido'],
        'rnehezseg' => $_POST['rnehezseg'],
        'uid' => $_SESSION['uid'],
    ];

    $apiUrl = 'http://recept24.szakdoga.net/api/recipeupload/index.php';

    $userAgent = 'Mozilla/5.0 (Windows NT 11.0; Win64; x64) AppleWebKit/537.36
    (KHTML, like Gecko) Chrome/99.0.9999.99 Safari/537.36';

    $ch = curl_init($apiUrl);

    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($data));
    curl_setopt($ch, CURLOPT_USERAGENT, $userAgent);

    $response = curl_exec($ch);
    $responseData = json_decode($response);

    if ($responseData[0]->hiba === 0) {

        echo "<h2>Sikeres receptfeltöltés!</h2>";

        echo '<a href="?p=upload">Feltöltök még egyet!</a>';

    } else {

        print($responseData[0]->hiba);

    }

}

if (!isset($_POST['rcim']) || (isset($_POST['rcim']) && ($responseData[0]->hiba != "0")))
{
    ?>
```

```

<h2 class="form-title">Recept Feltöltés</h2>

<form action="" method="post" id="recipe-upload-form" class="recipe-form">
  <div class="form-group">
    <label for="rcim" class="form-label">Recept Címe:</label>
    <input type="text" id="rcim" name="rcim" class="form-control" required>
  </div>

  <div class="form-group">
    <label for="rhozzavalok" class="form-label">Hozzávalók:</label>
    <textarea id="rhozzavalok" name="rhozzavalok" rows="4" class="form-control" required>
    </textarea>
  </div>

  <div class="form-group">
    <label for="rleiras" class="form-label">Leírás:</label>
    <textarea id="rleiras" name="rleiras" rows="8" class="form-control" required></textarea>
  </div>

  <div class="form-group">
    <label for="rido" class="form-label">Elkészítési Idő (perc):</label>
    <input type="number" id="rido" name="rido" class="form-control" required>
  </div>

  <div class="form-group">
    <label for="rnehezseg" class="form-label">Nehézség:</label>
    <select id="rnehezseg" name="rnehezseg" class="form-control" required>
      <option value="1">1</option>
      <option value="2">2</option>
      <option value="3">3</option>
      <option value="4">4</option>
      <option value="5">5</option>
      <option value="6">6</option>
      <option value="7">7</option>
      <option value="8">8</option>
      <option value="9">9</option>
      <option value="10">10</option>
    </select>
  </div>

  <input type="submit" value="Recept feltöltése" class="btn-submit">
</form>

<?php
}

?>

```

A folyamat két fő részből áll.

Az első rész a POST kérés feldolgozása, amely a recept adatait küldi a szerverre. Ha a `$_POST['rcim']` mező be van állítva, azaz ha a felhasználó elküldte az űrlapot, akkor előkészíti az adatokat a küldéshez. Ezután CURL segítségével POST kérést küld a megadott API végpontnak, amely a receptfeltöltést végzi. Ha a válaszban a hiba értéke 0, azaz ha a receptfeltöltés sikeres volt, akkor a kód gratulál a felhasználónak, és lehetőséget ad újabb recept feltöltésére. Ellenkező esetben, ha hiba történt, a kód megjeleníti a hibaüzenetet.

A második rész az űrlap megjelenítése, amely lehetővé teszi a felhasználók számára a recept adatainak megadását. Ha a `$_POST['rcim']` mező nincs beállítva, azaz ha a felhasználó még nem küldte el az űrlapot, vagy ha a receptfeltöltés során hiba történt, akkor megjeleníti az űrlapot a recept adatokkal, mint például a recept címe, hozzávalók, leírás, elkészítési idő és nehézség. A felhasználó kitölti az űrlapot, majd elküldi, hogy új receptet adjon hozzá a rendszerhez.

Különböző körülmények, esetek és hibakezelések

Keresési folyamat kezelése

Az alkalmazás keresési funkciója esetében fontos, hogy megfelelően kezeljük a felmerülő különböző körülményeket és hibákat mind a backend, mind a frontend részén.

A backend kódban először is figyelmet kell fordítanunk arra az esetre, amikor a felhasználó nem ad meg keresési paramétert, vagy üres értéket küld a GET kérésben. Ebben az esetben a backend válaszul egy hibaüzenetet küld vissza a frontendnek, amely figyelmezteti a felhasználót, hogy töltsse ki a keresőablakot. Ha a keresési paraméter megfelelő és talál találatokat az adatbázisban, akkor a backend JSON objektumban küldi vissza a talált recepteket a frontend számára, hogy megjeleníthesse azokat. Amennyiben a keresési paraméterrel nem található recept az adatbázisban, a backend egy megfelelő hibaüzenetet küld vissza a frontendnek, hogy tájékoztassa a felhasználót arról, hogy nincs találat.

A frontend kódban a hiányzó keresőmezőt kezeljük először. Ha a felhasználó nem tölti ki a keresőmezőt és elküldi a formot, a frontend hibaüzenetet jelenít meg, hogy kérje a felhasználót, hogy töltsse ki a keresőmezőt. Ha érvénytelen adatot ad meg a felhasználó a keresőmezőben, a frontend végzi a validációt és megjeleníti a hibaüzenetet az érvénytelen adatokról. Ha a frontend nem tud kapcsolódni a backendhez az adatok lekérdezésekor, akkor megjeleníti a hálózati hibaüzenetet a felhasználónak, és javasolja, hogy próbálkozzon később újra.

Belépési folyamat kezelése

A belépési folyamat során fontos megfelelően kezelni az esetleges különböző körülményeket és hibákat mind a backend, mind a frontend részén. Elsőként figyelembe kell venni azt az esetet, amikor a felhasználó nem tölti ki a felhasználónév vagy jelszó mezőt. Ebben az esetben a frontend hibaüzenetet jelenít meg, hogy kérje a felhasználót, hogy töltsse ki mindkét mezőt. Amennyiben a felhasználó megadja a helyes felhasználónevet és jelszót, a backend ellenőrzi az adatokat az adatbázisban. Ha a bejelentkezés sikeres, akkor a backend JSON objektumban visszaküldi a felhasználó adatait a frontendnek, amely megjeleníti a sikeres bejelentkezést. Abban az esetben, ha a felhasználó hibás felhasználónevet vagy jelszót ad meg, a backend hibaüzenetet küld vissza a frontendnek, hogy tájékoztassa a felhasználót a hibás bejelentkezési adatokról. Ha a frontend nem tud kapcsolódni a backend szerverhez a bejelentkezési adatok elküldésekor, akkor hibaüzenetet jelenít meg a felhasználónak, hogy tájékoztassa a kapcsolódási problémáról, és javasolja, hogy próbálkozzon később újra. Végül, ha a felhasználó sikeresen bejelentkezik, a frontend átirányítja őt a főoldalra, hogy folytathassa a rendszer használatát.

Fejlesztési lehetőségek

A modern gasztronómia világában elengedhetetlen egy olyan receptes weboldal, amely egyszerűen és inspirálóan szolgál mindazok számára, akik szenvedélyesen szeretnek főzni és megosztani élményeiket. Ebben a fejlesztésben rengeteg lehetőség rejlik, melyek segítségével még vonzóbbá és hatékonyabbá tehetjük az oldalt.

Példaként néhány fejlesztési lehetőség:

- **Képek és videók hozzáadása a receptekhez:** Ez lehetővé teszi a felhasználóknak, hogy még érthetőbben megosszák receptjeiket, bemutassák az elkészítés folyamatát, és inspiráljanak másokat.
- **Receptek kategorizálása:** Segítse a felhasználókat a könnyebb böngészésben és a receptek megtalálásában a kategóriák használatával, például vegetáriánus, gyorsétel stb.
- **Receptkönyv funkció:** Kínáljon lehetőséget a felhasználóknak, hogy saját receptkönyvet hozzanak létre, így könnyen elérhetik és megosszák kedvenc ételeiket.

- **Értesítések és tevékenységek követés:** Biztosítsa a felhasználóknak az értesítéseket az új receptekről, kommentekről vagy követett felhasználók aktivitásáról, így folyamatosan kapcsolatban maradhatnak az oldallal.
- **Chat funkció:** Nyújtson lehetőséget a felhasználók számára a közvetlen kommunikációra, recepttanácsok megosztására vagy közös sütés-főzés tervezésére egy chat ablakon keresztül.

Üdvözöllek!

Kézben tartod a Recept24 weboldalát, egy online receptmegosztó platformot, ahol könnyedén felfedezhetsz új gasztronómiai ötleteket és megoszthatod saját receptjeidet.

A Recept24 alapvető funkciói között szerepel a belépés és regisztráció lehetősége. Ha még nem regisztráltál, egyszerűen létrehozhatod a fiókot, majd bejelentkezhetsz az oldalra, hogy teljes körű hozzáférést kapj a funkciókhoz.

Emellett a weboldalon lehetőség van receptek böngészésére és keresésére. Fedezz fel új ízletes fogásokat, böngéssz különböző kategóriák szerint, vagy használd a keresőt a kedvenc receptjeid gyors megtalálásához.

Ha saját recepttel rendelkezel, könnyedén megoszthatod azt a közösséggel. Egyszerűen töltsd fel a recept címét, hozzávalóit, leírását, elkészítési időt és nehézségi szintet, hogy mások is élvezhessék az általad alkotott ínycsiklandozóságot.

A Recept24 segítségével könnyedén inspirálódhatsz a konyhában, új fogásokat próbálhatsz ki, és tapasztalatokat cserélhetsz az ételkedvelő közösséggel.

Köszönjük, hogy csatlakoztál hozzánk, és jó étvágyat kívánunk az ízletes kalandokhoz!

Üdvözlettel,

Recept24 csapata

Szükséges eszközök a Recept24 weboldal használatához

A weboldalhoz való hozzáféréshez és használatához csak egy internetkapcsolattal rendelkező eszközre van szükség, például:

- Számítógép (laptop vagy asztali számítógép)
- Okostelefon
- Táblagép

Ezen eszközök bármelyike alkalmas a Recept24 weboldalának böngészésére, receptek keresésére és feltöltésére.

Weboldal eléréséhez használható böngészők

Az internetkapcsolat mellett szükség lesz egy modern webböngészőre, például:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

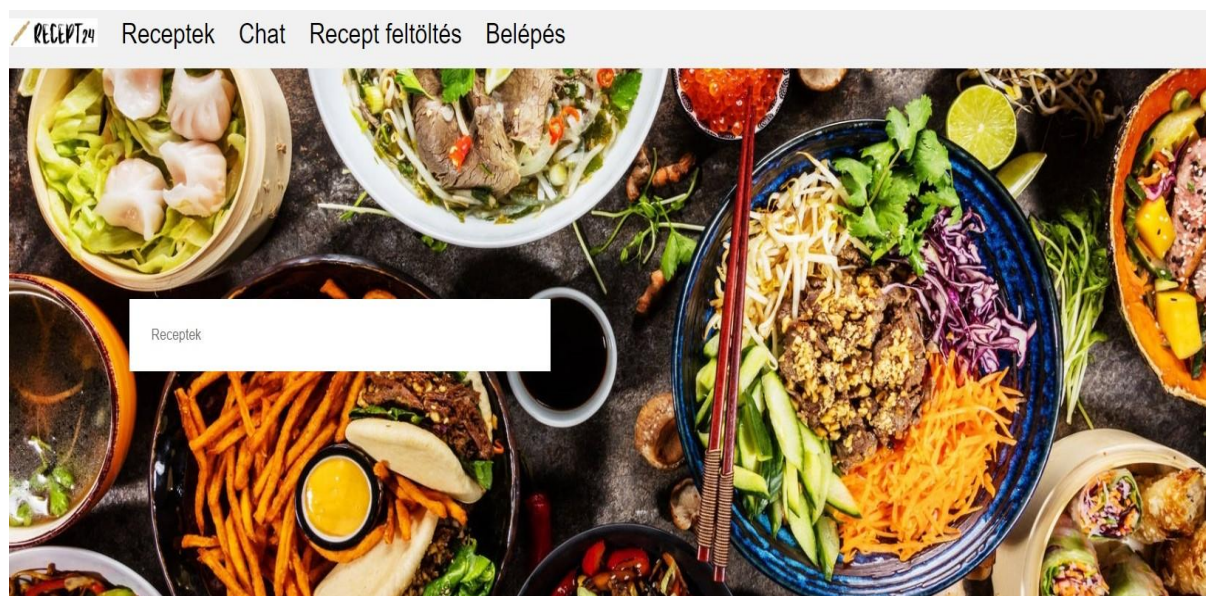
Ezek a böngészők lehetővé teszik a Recept24 weboldalának teljes körű használatát és böngészését.

Weboldal elérhetősége

A Recept24 weboldalhoz a következő módon juthatsz hozzá:

1. Nyisd meg a kedvenc webböngésződet (pl. Google Chrome, Mozilla Firefox).
2. Az URL sorba írd be a Recept24 weboldal címét:
recept24.szakdoga.net

Weboldal használatának részletes ismertetése

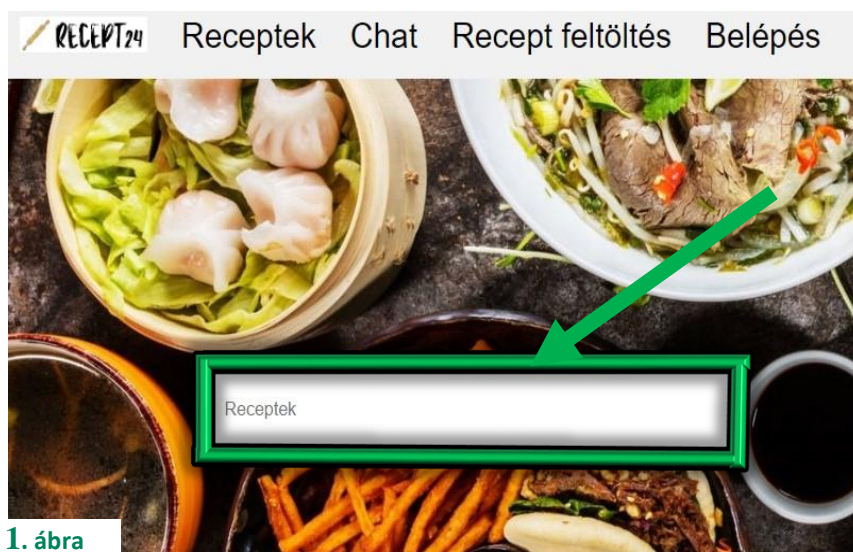


A Recept24 főoldalát látod, itt megtalálod a  **RECEPT24** logóját, Receptek, Chat, Recept feltöltés, Belépés gombokat.

- A logóra kattintva mindig visszatérhetsz a főoldalra, bármelyik másik oldalon is jársz éppen.
- Tekintsd át a közösség által megosztott receptek széles választékát a "Receptek" gombra kattintva.
- A 'Chat' gomb segítségével hamarosan csatlakozhatsz az élő csevegéshez, ahol más felhasználókkal oszthatod meg tapasztalataidat és recept ötleteidet. Kérjük, addig is tarts velünk, hamarosan elérhető lesz ez a funkció.
- Ha saját recepttel rendelkezel és szeretnéd megosztani, kattints a "Recept feltöltés" gombra. Fontos, hogy ehhez először regisztrálj vagy jelentkezz be a fiókodba!
- A "Belépés" gombra kattintva lehetőséged van belépni a fiókodba vagy új fiókot létrehozni, hogy teljes körű hozzáférést kapj az oldal szolgáltatásaihoz.

A szabadon elérhető szolgáltatásaink közé tartozik a receptek böngészése és keresése. Amennyiben saját receptet szeretnél megosztani, regisztrálnod kell a recept feltöltéséhez.

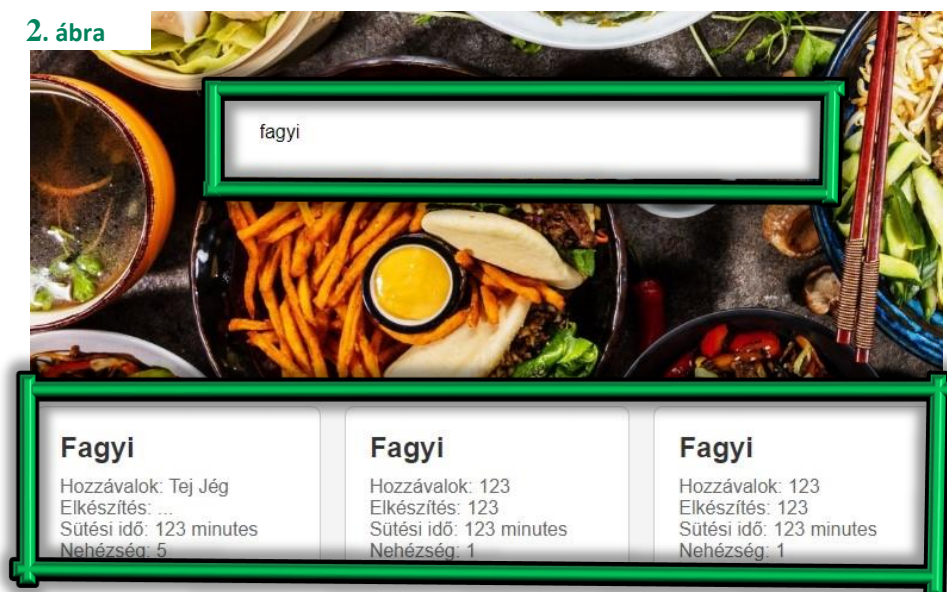
Recept keresés folyamatának ismertetése



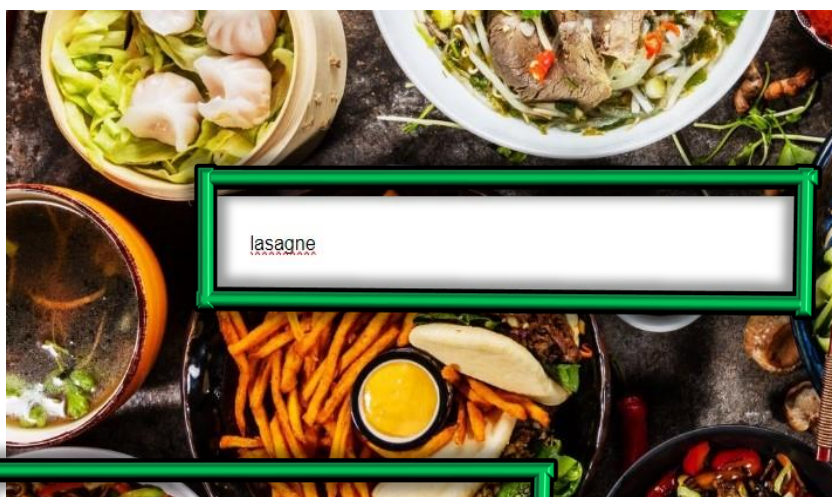
1. ábra

Az **első** ábrán látható, hogy a nyíllal jelölt területen tudod megkeresni a recepteket. Csak az alapvető keresési funkciót látod, és nincsenek speciális keresési lehetőségeid vagy szűrőid. A keresési eredményeket nem lehet rendezni vagy tovább szűrni a beállításaid alapján. Most az a lényeg, hogy könnyen és gyorsan megtaláljátok azokat a recepteket, amik érdekelnek titeket. Ezek az alapvető funkciók az oldal felhasználói barátságos maradását biztosítják, és segítenek nektek gyorsan megtalálni, amit keresel.

2. ábra



A **második** ábrán a keresőmezőbe beírtam a keresett recept nevét, majd az enter gomb lenyomásával a rendszer listázza az összes találatot. Ez a funkció lehetővé teszi, hogy pontosan azt a receptet találj meg, amit keresel, és gyorsan hozzáférj az információkhoz.

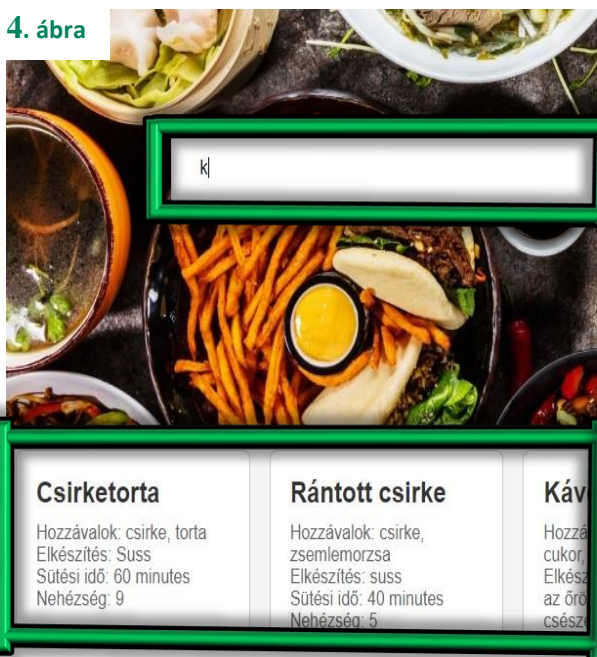


5. ábra

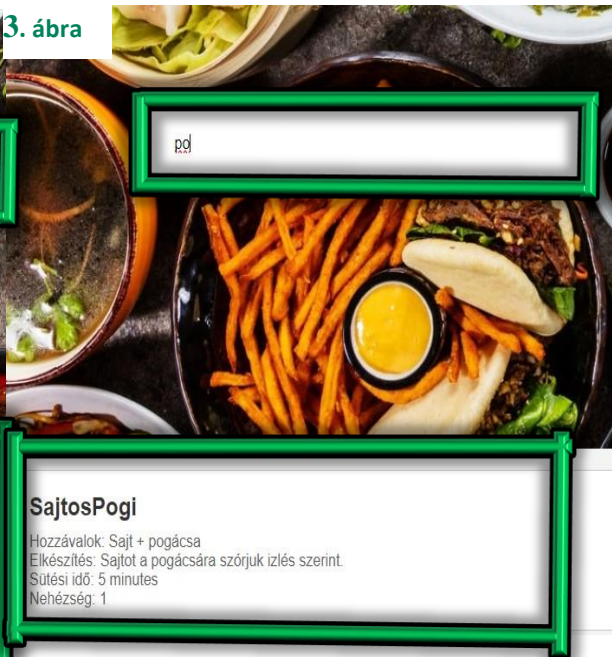
A **harmadik** ábrán látható, hogy egy konkrét receptre keresve nem találtam találatot. Ez azt jelenti, hogy vagy nem létezik ilyen nevű recept az adatbázisban, vagy a keresett recept névhibásan lett megadva. Ilyen esetben a rendszer egy megfelelő hibaüzenetet küld vissza,

tájékoztatva téged arról, hogy nem talált ilyen nevű receptet. Fontos megjegyezni, hogy ha érvénytelen vagy helytelen karaktereket írsz be a keresőmezőbe, ugyanazt a hibaüzenetet fogod kapni.

4. ábra



3. ábra



A **négyes** és az **ötös** ábrán látható, hogy már egy, illetve két karakter beírásával is recepteket találunk, mivel a keresett karakterek szerepelnek a receptek címében. Ez a funkció lehetővé teszi, hogy még akkor is eredményeket kapj, ha csak részleges információt adsz meg a keresőmezőben. Fontos tudnod, hogy ez a keresési mód nem specifikus, és több találatot adhat, amelyek között válogatnod kell majd. Így könnyebben megtalálhatod azokat a recepteket, amik érdekelnek, még akkor is, ha csak egy vagy két karaktert adsz meg a keresőmezőben.

Regisztráció folyamatának ismertetése

A zöldel bekeretezett részek segítenek a regisztráció eljutásához

1. ábra



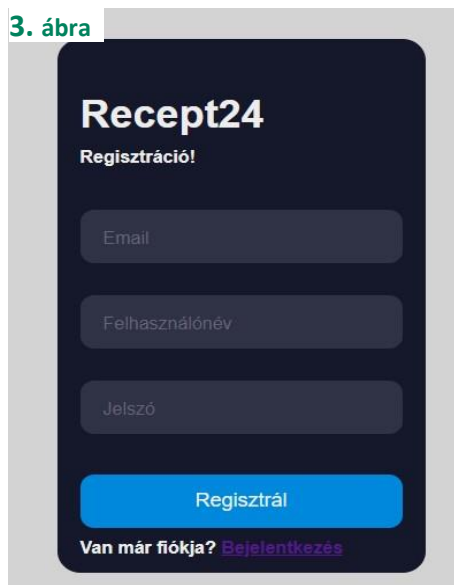
Az **első** ábrán láthatók a gombok. A belépés gombra kattintva jutsz el a regisztrációhoz.

2. ábra



A **második** ábrán látható a bejelentkezési felület. Alul, a 'Még nem regisztráltál?' felirat mellett találod a 'Regisztráció' gombot, amire rákattintva jutsz el a regisztrációhoz.

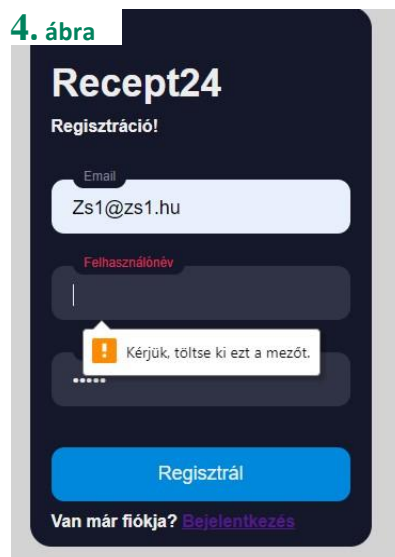
3. ábra



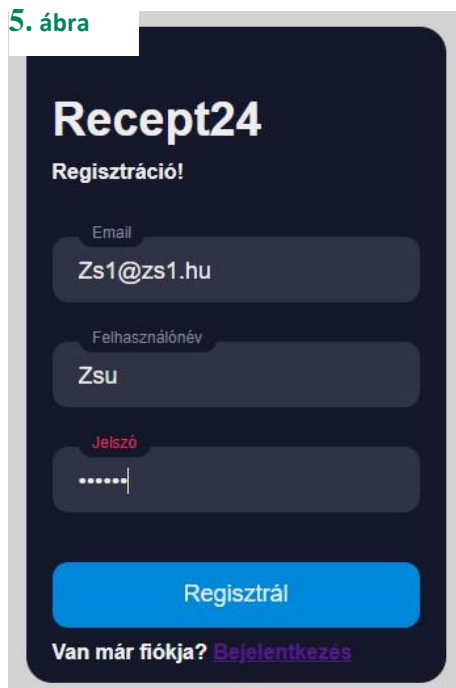
A **harmadik** ábrán látható a regisztráció. Itt megadhatod az e-mail címedet, felhasználónevedet és jelszavadat. Végül a 'Regisztrál' gombra kattintva regisztrálhatsz.

A **negyedik** ábra azt mutatja, hogy ha bármelyik mező nincs kitöltve, például az email cím, felhasználónév vagy jelszó, akkor nem engedi a regisztrációt, és ezt a hibaüzenetet kapod: "Kérem, töltsé ki az összes mezőt."

4. ábra



5. ábra



Az **ötödik** ábrán láthatod, hogy kitöltöttem minden adatot: e-mail címet, felhasználónevet és jelszót, majd rányomtam a 'Regisztráció' gombra.

6. ábra



Az **hatodik** ábrán megtekintheted, hogy milyen üzenet fogad, ha sikeresen regisztráltál. Ez a lépés megerősíti, hogy regisztrációd sikeres volt, és felkészít arra, hogy beléphess a weboldalra az új fiókkal.

A **hetedik** ábrán láthatod, hogy ha már ugyanolyan e-mailcímmel vagy



7. ábra

felhasználónévvel próbálsz regisztrálni, mint ami már létezik a rendszerben, akkor a regisztráció nem lehetséges. A hibaüzenet arra kér, hogy adj meg egy másik e-mailcímet vagy felhasználónevet a regisztrációhoz. Ez segít megelőzni a konfliktusokat és biztosítja a felhasználók egyediségét a rendszerben.

A regisztráció részletes bemutatása során megismerhettük, hogyan juthatsz el a regisztrációs felületig és hogyan tudsz sikeresen regisztrálni. Emellett áttekintettük a lehetséges hibaüzeneteket is, amelyek segítenek az esetleges problémák azonosításában. Most már készen állsz arra, hogy csatlakozz a közösségünkhöz és kihasználd az összes rendelkezésre álló lehetőséget!

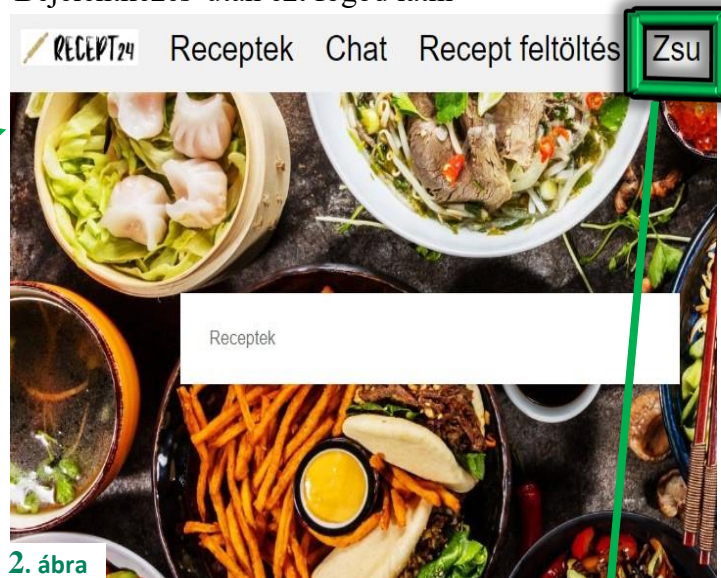
Belépés folyamatának ismertetése

Az előző regisztrációs folyamat során már bemutattam, hogy hogyan juthatsz el a bejelentkezési felülethez. Ehhez egyszerűen kattints a "Belépés" gombra, amely lehetővé teszi, hogy a bejelentkezési oldalra navigálj. Ez az opció lehetővé teszi számodra, hogy már meglévő fiókkal jelentkezz be, így hozzáférhetsz a weboldalunk teljes funkcionalitásához

1. ábra

Az **első** ábrán megjelenik a bejelentkező felület, ahol először is a korábban regisztrált felhasználónevedet, majd a hozzá tartozó jelszavadat kell megadnod. A 'Bejelentkezés' gombra kattintva jutsz el a fiókodba.

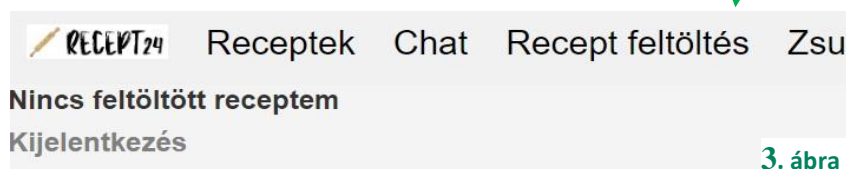
'Bejelentkezés' után ezt fogod látni



2. ábra

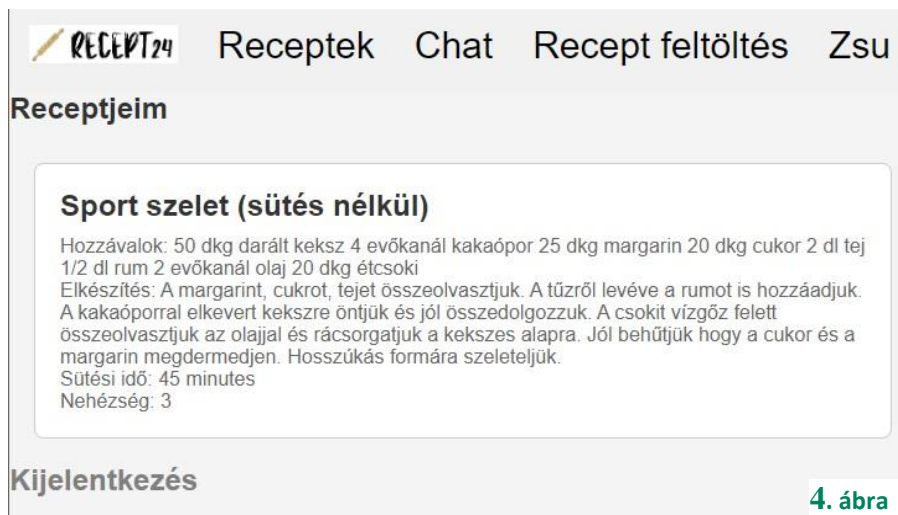
A **második** ábrán miután sikeresen bejelentkeztél, azonnal átkerülsz a főoldalra, ahol a felhasználónevedre kattintva további funkciókhoz férhetsz hozzá.

'Felhasználónevedre' kattintva fogod látni



3. ábra

A **harmadik** ábrán felhasználónevedre kattintva megjelenik egy menü, ahol láthatod a "Nincs feltöltött receptem" és a "Kijelentkezés" lehetőségeket. Ezek az opciók segítenek abban, hogy könnyen kezelhesd a receptjeidet és ki tudj lépni a fiókból, ha szükséges.



A **negyedik** ábrán a felhasználónevedre kattintva egy menü jelenik meg, ahol már láthatod a "Receptjeim" és a "Kijelentkezés" lehetőségeket. Ez az opció a korábbiakhoz képest azt jelzi, hogy már feltöltöttél recepteket, így könnyen hozzáférhetsz.

A belépés folyamatának bemutatása során megismertük, hogyan juthatsz el a bejelentkező felületig, és hogyan tudod megadni a korábban regisztrált felhasználóneved és jelszavad. Emellett áttekintettük, hogy milyen lehetőségeket kínál a rendszer a bejelentkezés után, és hogyan navigálhatsz az egyes funkciók között. Most már készen állsz arra, hogy belépj a rendszerbe, és teljes mértékben kihasználj az előnyeit!

Recept feltöltés folyamatának ismertetése

Fontos megjegyezni, hogy amikor receptet szeretnél feltölteni, először be kell jelentkezned. Ezt könnyen észre fogod venni, amikor a weboldal főoldalán vagy, és rákattintasz a "Recept feltöltés" gombra. Ekkor megjelenik a bejelentkező lap, ahol be kell jelentkezned, vagy ha még nem rendelkezel fiókkal, akkor regisztrálnod kell. Ha már be vagy jelentkezve, akkor pedig közvetlenül megjelenik a "Recept feltöltés" menüpont.

1. ábra **Recept Feltöltés**

Recept Címe:

Sport szelet (sütés nélkül)

Hozzávalók:

20 dkg cukor
2 dl tej
1/2 dl rum
2 evőkanál olaj
20 dkg étcsoki

Leírás:

A margarint, cukrot, tejet összeolvasztjuk. A tűzről levéve a rumot is hozzáadjuk. A kakaóporral elkevert keksze öntjük és jól összedolgozzuk. A csokit vízgőz felett összeolvasztjuk az olajjal és rácsorgatjuk a kekszes alapra. Jól behűtjük hogy a cukor és a margarin megdermedjen. Hosszúcs formára szeleteljük.

Elkészítési Idő (perc):

45

Nehézség:

3

Recept feltöltése

Az **első** ábrán látható a "Recept Címe", "Hozzávalók", "Leírás", "Elkészítési idő (perc)", "Nehézség" mezők. Ezen mezők kitöltésével a "Recept feltöltése" gombra kattintva fel is töltöttül a receptünket.

Sikeres receptfeltöltés!

[Feltöltők még egyet!](#)

2. ábra


A **második** ábra a "Sikeres receptfeltöltés" felületét mutatja, amelyet előzőleg már bemutattam a belépésnél. Ha a "Felhasználónév" gombra kattintasz, megtalálod az általad feltöltött receptet.

3. ábra **Recept Feltöltés**

Recept Címe:

Hozzávalók:

Leírás:

 Kérjük, töltsd ki ezt a mezőt.

Elkészítési Idő (perc):

Nehézség:

A **harmadik** ábra azt mutatja, hogy ha bármelyik mező nincs kitöltve, például a Recept címe, Hozzávalók, leírás vagy elkészítési idő akkor nem engedi a recept feltöltést, és ezt a hibaüzenetet kapod: "Kérjük, töltsd ki ezt a mezőt."

A recept feltöltés folyamatát bemutatva megismertük, hogyan tudsz eljutni a receptfeltöltő felületig, és hogyan tudod feltölteni az új recepteket. Emellett áttekintettük, hogy milyen lehetőségeket kínál a rendszer a receptfeltöltés után, és hogyan tudsz navigálni az egyes funkciók között. A rendszer hibaüzenetekkel is ellát, amelyek segítséget nyújtanak az esetleges problémák megoldásában. Most már készen állsz arra, hogy használd a receptfeltöltés funkcióját, és megoszd saját receptjeidet a közösséggel!

Elkészült munkánk értékelése

Rendkívül elégedettek vagyunk a csapatunk teljesítményével! Sikerült egy szuper alapot kiépítenünk a további fejlesztésekhez, és nagyon büszkék vagyunk erre. A weboldalunk teljesíti az összes elvárásunkat, és reméljük, hogy ez az erőfeszítés hosszú távon is hozzájárul majd a projekt sikeréhez.

Sokat tanultunk és fejlődünk, bár voltak kihívások, ezek mind segítettek minket a további javulás felé. Eredetileg egy könnyen kezelhető receptes oldalt akartunk létrehozni, ahol mindenki könnyen találhat és megoszthat recepteket, néhány dologban még nem vagyunk tökéletesek. Például, a chat funkció még nincs teljesen kész, és az egyedi profilok létrehozása sem volt mindenki számára tökéletesen elérhető. Ezek a kihívások további lehetőségeket kínálnak számunkra a fejlődésre, és biztosak vagyunk benne, hogy a jövőben még jobbra tehetjük az oldalt!

Sok ötletünk van arra, hogyan tehetnénk még jobbra és vonzóbbá az oldalt.

Először is, tervezünk további funkciók bevezetését, mint például a chat funkció teljes körű működését és a felhasználói profilok testreszabhatóságát. Ezek az újítások lehetővé teszik majd a felhasználók számára, hogy még jobban kihasználják az oldalunkat és könnyebben kommunikáljanak egymással. Emellett folytatni fogjuk a tartalomfejlesztést is, például új receptekkel és hasznos tanácsokkal. Ez segíthet abban, hogy az oldalunk még vonzóbbá váljon az ételkedvelők számára és növelje a visszatérő látogatók arányát. Fontosnak tartjuk továbbá az oldalunk felhasználói visszajelzéseinek figyelembevételét és az igények alapján történő folyamatos fejlesztést. Ez segíthet abban, hogy az oldalunk valóban azon igényeket elégítse ki, amelyekre a felhasználóinknak szükségük van. Bízunk benne, hogy ezek a fejlesztések hozzájárulnak majd az oldalunk további sikeres hasznosulásához és növekedéséhez.

Köszönetnyilvánítás

Elsősorban szeretnénk köszönetet mondani családjainknak a mindennapi támogatásért és szeretetért. Ők voltak azok, akik mindig mellettünk álltak, és akiktől erőt merítettünk a nehéz pillanatokban.

Köszönet illeti mindazokat a tanárokat, akik az elmúlt két évben tanítottak és segítettek bennünket az úton. A velük töltött idő alatt nemcsak tudásban, hanem életre szóló tanulságokban is részesültünk.

Végül, de nem utolsósorban, szeretnénk kifejezni hálánkat Kovács László Tanár úrnak a sok-sok segítségért, tanításért, tanácsért és ötletért, amit nekünk adott. Ő mindig mellettünk állt, és bátorított bennünket a fejlődésben és a kitartásban.

<https://www.w3schools.com/>

<https://webiskola.hu/>

<https://infojegyzet.hu/>