

# **Záródolgozat**



**Kis Tímea**

**Szoftverfejlesztő és -tesztelő**

**5/13/A**

**Kecskeméti SZC**

**Kandó Kálmán Technikum**

**2024**

# Tartalomjegyzék

<b>Bevezetés</b> .....	3
Bemutakozás .....	3
Ötlet.....	3
<b>Felhasználói Dokumentáció</b> .....	4
Rendszerkövetelmény .....	4
Elérhetőség .....	4
Felhasználói Felület Bemutatása .....	5
Főoldal.....	5
Keresés .....	6
Recept Megtekintése .....	7
Bejelentkezés/Regisztráció.....	8
Profil.....	9
Recept feltöltése .....	10
<b>Fejlesztői Dokumentáció</b> .....	11
Használt Technológiák És Eszközök.....	11
Adatbázis Felépítése.....	12
Adatbázis Kapcsolati Diagram.....	15
Fájlrendszer .....	16
Programozási Technikák .....	17
RestAPI szoftverarchitektúra .....	17
API .....	17
PDO .....	19
Hibakezelés .....	19
Preparált Lekérdezések és Paraméterek Kötése .....	19
Hash.....	20
Munkamenet Kezelése .....	20
Dinamikus Tartalomgenerálás .....	23
További Fejlesztési Tervek.....	26
<b>Források</b> .....	27

# Bevezetés

## Bemutakozás

Weboldalam egy olyan receptmegosztó közösségi tér, ahol minden korosztály és konyhai tudásszint megtalálhatja a helyét – legyen szó a mindennapi gyors ételekről vagy az ünnepi asztalra való különlegességekről.

Funkciói között szerepel a széleskörű receptkeresés, amely lehetővé teszi, hogy bárki könnyedén rátaláljon a kívánt ételre.

A regisztrált felhasználók számára pedig további lehetőségek nyílnak meg: saját receptek feltöltése, mások által megosztott receptek értékelése, mentése.

## Ötlet

A Flavora weboldal ötlete saját mindennapi szükségleteim és tapasztalataim alapján született meg. A cél egy olyan oldal megalkotása volt, amely megkönnyíti hétköznapijainkat és nap mint nap hasznos lehet számunkra. Emellett fontos volt számomra, hogy egy olyan felületet hozzak létre, ahol könnyen megtalálhatja a felhasználó az elkészítendő receptet.

# Felhasználói Dokumentáció

## Rendszerkövetelmény

A látogatóknak csupán internetkapcsolatra van szükségük, valamint egy webböngészőre, ami lehetővé teszi a weboldalak megjelenítését, mint például a Google Chrome, Mozilla Firefox vagy Safari.



Ezáltal a weboldal operációs rendszertől függetlenül, széles körű eszközkompatibilitást biztosít, beleértve a mobiltelefonokat, tableteket, laptopokat és asztali számítógépeket.

## Elérhetőség

Weboldalam a <https://flavora.hu/> címen érhető el.

Az oldalam könnyebb elérhetősége érdekében, még a kevésbé pontos címek, mint "http://flavora.hu", "http://www.flavora.hu" és "https://www.flavora.hu", is a biztonságos "https://flavora.hu" webcímre irányítódnak át automatikusan egy állandó (301-es) átirányítás segítségével. Ez a folyamat biztosítja, hogy a látogatók a lehető legbiztonságosabban érhék el az oldalt, függetlenül attól, hogy melyik címet használják a kezdeti hozzáférés során.

# Felhasználói Felület Bemutatása



1. ábra: Navigációs sáv

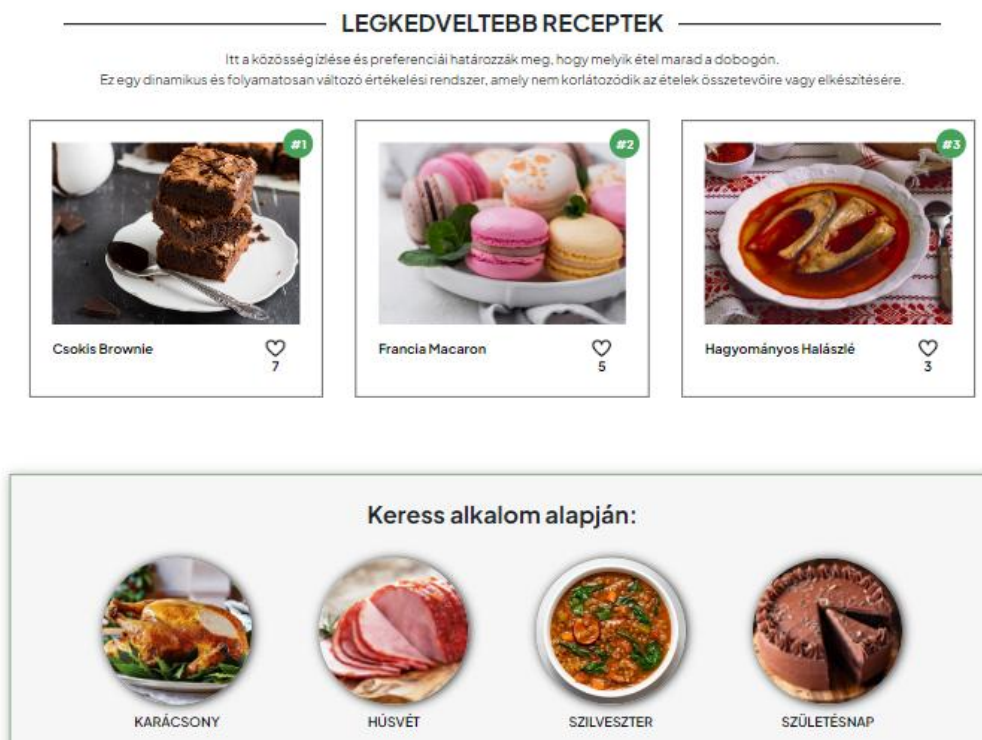
A weboldal kulcseleme a navigációs sáv, amely állandóan a lap tetején helyezkedik el. Ez a sáv tartalmazza a weboldal legfontosabb funkcióit, amelyek elősegítik a felhasználók számára az egyszerű navigálást és a receptek gyors megtalálását.

Továbbá, a navigációs sáv adaptív kialakításának köszönhetően a felület könnyen kezelhető marad mobil eszközökön is. Mobil nézetben a navigációs elemek egy lenyíló menübe rendeződnek, ami optimalizálja a képernyőterület használatát és a felhasználói élményt.

## Főoldal

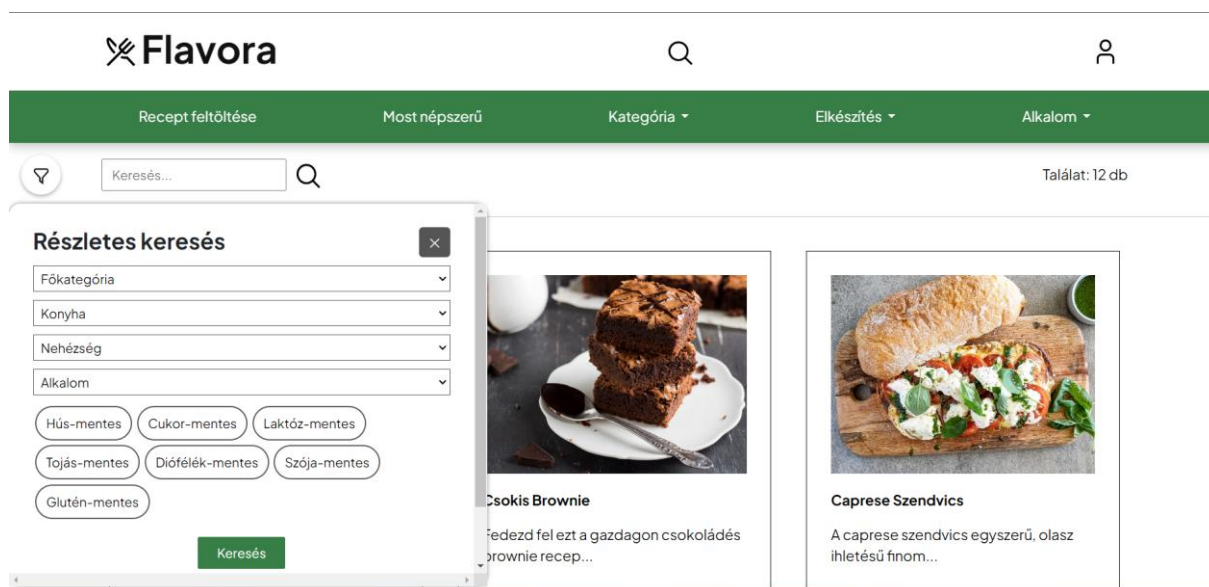
A „Flavora” felirat hivatkozásként szolgál a weboldal főoldalára, ide kattintva bárholnan visszatérhetünk a <https://flavora.hu/> -oldalra.

A „Most népszerű” szekció is a főoldalon található, amelyet a navigációs sáv alsó menüből érhetünk el; itt a felhasználók által legkedveltebb három recept kerül megjelenítésre.



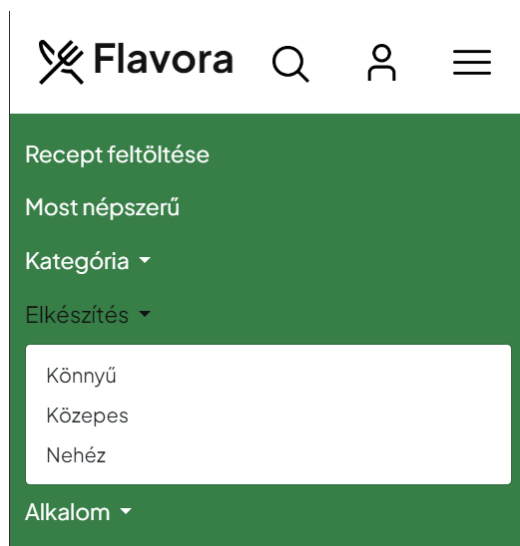
2. ábra: Részlet a főoldal tartalmáról

## Keresés



3. ábra: <https://flavora.hu/page/search>

A keresés funkció a részletes keresőoldalra vezet, ahol kezdetben minden receptet láthatunk szűrés nélkül. Itt lehetőségünk van kategóriák szerinti szűrésre, a receptek nevében való szöveges keresésre, allergének szerinti keresésre, és akár egyszerre több szempont alapján is finomíthatjuk a találatokat.



4. ábra: Navigációs sáv almenü mobil nézetből

A navigációs sáv „Kategória”, „Elkészítés”, „Alkalom” almenüi lehetővé teszik, hogy egyetlen kattintással szűrjük a recepteket. Például, ha az „Elkészítés” alatt a „Könnyű” opciót választjuk, ez automatikusan beállítja a „könnyű” nehézségi szűrőt, és átirányít minket a keresés oldalra. pl. ezen a címen:


<https://flavora.hu/page/search&difficulty=26>

## Recept Megtekintése

### Csokis Brownie

DesszertSüteményAmerikaiKönnyű





Fedezd fel ezt a gazdagon csokoládés brownie receptet, ami tökéletes desszert minden alkalomra. Kívül ropogós, belül puha és ragacsos - ez a sütemény garantáltan elvarázsol minden csokirajongót.



ktimi - 2024-01-28 7

Hozzávalók 6 adaghoz

Elkészítési idő: 60 perc



### Hozzávalók:

- 200 gramm étcsokoládé (min 70% kakaó)
- 175 gramm vaj

### Elkészítés:

1. Melegítsd elő a sütőt 180°C-ra (légkeveréses esetén 160°C). Vegyél elő egy kb. 20x20 cm-es

5. ábra: Recept megtekintése

Amennyiben a recept előnézeti képére kattintunk, a recept részletes adatlapja jelenik meg, ahol minden szükséges információt megtalálunk a recept elkészítéséhez szükséges lépésektől kezdve a további fontos adatokig, mint például hozzávalók, allergének, a recept készítőjének neve és a feltöltés dátuma. Ezen az oldalon van lehetőségünk a receptet kedvelni is, ehhez csupán a szív ikonra kell kattintanunk, bejelentkezés után.

A feltöltő nevére kattintva átnavigál a felhasználó profiljára, ahol az általa feltöltött többi receptet is megtekinthetjük, valamint az általa kedvelt recepteket is.

## Bejelentkezés/Regisztráció

Flavora

BEJELENTKEZÉS **REGISZTRÁCIÓ**

TELJES NÉV

FELHASZNÁLÓNÉV

JELSZÓ

JELSZÓ ISMÉT

EMAIL CÍM

Neme:

Nő

REGISZTRÁCIÓ

A név megadása kötelező! A felhasználónév megadása kötelező! Az email megadása kötelező! A jelszó megadása kötelező!

6. ábra: Regisztráció + hibaizenet




A regisztrációs folyamat részeként a felhasználóktól azt is kérjük, hogy adják meg nemüket, amely információ szükséges a profilkép megfelelő beállításához. Annak érdekében, hogy megóvjuk a felhasználóink személyes adatait és elkerüljük az illetlen képek feltöltését, a profil képének kiválasztását két előre definiált avatárra korlátozzuk a nem alapján.



7. ábra: Profilképek




## Profil



Most népszerű

Kategória ▾

Elkészítés ▾




**Kis Tímea**  
@ktimi  
regisztrált: 2024-02-18


**3**  
feltöltött recept


**3**  
kedvelt recept

**FELTÖLTÖTT RECEPTEK**





Bolognai Spagetti Egyszerűen






Csokis Brownie








Caprese Szendvics



**KEDVELT RECEPTEK**







8. ábra: Profil oldal

Bejelentkezés után ezen a felületen megtekinthetjük profilunk adatait.

A profiloldalon lehetőségünk nyílik áttekinteni az általunk feltöltött recepteket és a kedvencnek jelölt recepteket. A kedvelés lehetővé teszi, hogy egy gyors és egyszerű módon gyűjtsük össze azokat a recepteket, amelyeket kipróbálnánk a jövőben. Az elmentett recepteket bármikor megtekinthetjük, így amikor inspirációt keresünk a következő étkezéshez, egyszerűen visszatérhetünk kedvenceinkhez és a később elkészíteni kívánt receptekhez.

Saját receptjeinket innen tudjuk törölni, az adott recept alatt található gombbal.

A korábbi „profil” gomb ekkor „kijelentkezés” gombra változik.

## Recept feltöltése

Ezen az oldalon lehetősége van a bejelentkezett felhasználóknak, hogy saját receptjeiket könnyedén és gyorsan osszák meg egy felhasználóbarát felületen keresztül.

A receptek megosztása az oldal legfőbb funkciója, ezért arra törekedtem, hogy a feltöltési folyamatot a lehető legkönnyebbre és leggyorsabbra tervezzem.

Célom, hogy a felhasználók számára minél kevesebb teherrel járjon a receptmegosztás, ezért számos funkciót vezettem be, amelyek megkönnyítik a receptek feltöltését.

Ilyen például a képfeltöltési útmutató, amely praktikus tanácsokat nyújt a fotók elkészítéséhez. Valamint a felhasználók könnyedén választhatnak a különböző ételkategóriák között, az előre megadottak közül.

A mezőkbe előre beírt placeholdererek konkrét példákat adnak arra vonatkozóan, hogyan töltsék ki az adott mezőt.

A hozzávalók megadásakor a rendszer különösen felhasználóbarát módon segít a felhasználóknak. Már egyetlen betű beírásával is az adatbázisunkban található hozzávalók listája jelenik meg, lehetővé téve a gyors kiválasztást. Ha a felhasználó által keresett hozzávaló nem szerepel az adatbázisunkban, egyszerűen beírhatják azt, és az új hozzávaló automatikusan bekerül az adatbázisba, így legközelebb már az is elérhető lesz más felhasználók számára is.

A mértékegységek megadása még egyszerűbbé válik, hiszen ez egy legördülő listából történik, amely már tartalmazza az összes lehetséges opciót.

Ez a megközelítés garantálja, hogy a receptek feltöltése során a mértékegységek egységesek és pontosak maradnak, tovább egyszerűsítve a felhasználók számára a receptmegosztás folyamatát.

The screenshot displays the 'HOZZÁVALÓK' (Ingredients) section of a recipe upload form. It features a search input field containing the letter 's', which has triggered a dropdown list of suggestions: 'só', 'sonka', 'spagetti tészta', 'sűrített paradicsom', 'sütőpor', 'szalonna', and 'szódabikarbóna'. To the right of the search field is a 'Mennyiség:' (Quantity) field set to '500'. Further right is a 'Mértékegység:' (Unit) dropdown menu, currently showing 'centigramm' as the selected option. Other units visible in the list include gramm, milligramm, dekagramm, kilogramm, gallon, liter, milliliter, deciliter, centiliter, csepp, csomag, szem, kis darab, and darab. A green button labeled 'Hozzávaló hozzáadás' (Add ingredient) is positioned at the bottom left of the form area.

9. ábra: Részlet a recept feltöltésről

# Fejlesztői Dokumentáció

## Használt Technológiák És Eszközök

### Visual Studio Code



Visual Studio Code-ot használtam, mivel ez egy rendkívül rugalmas kód szerkesztő, amely támogatja a különféle programozási nyelveket és technológiákat.

### HTML



HTML5-öt alkalmaztam a weboldal alapvető szerkezetének és tartalmi elemeinek megalkotására, biztosítva ezzel a weboldal szemantikus felépítését.

### CSS



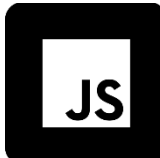
CSS3-at használtam a weboldal vizuális megjelenésének finomhangolására, lehetővé téve, hogy a weboldal esztétikailag vonzó és könnyen olvasható legyen minden látogató számára.

### Bootstrap



A Bootstrap 5.0.2 használatával könnyedén implementáltam reszponzív dizájnt, amely alkalmazkodik a különböző kijelző méretekhez és eszközökhöz.

### JavaScript



JavaScript segítségével adtam hozzá dinamikus funkciókat, mint az interaktív menüket, felhasználói űrlapok validálását, ezzel növelve a weboldal interaktivitását és felhasználói élményét.

### PHP



PHP-t használtam a szerveroldali logika implementálására, beleértve az adatok kezelését, tartalmak generálását, és a felhasználói interakciókat.

### MySQL



MySQL adatbázist alkalmaztam az adatok tárolására és kezelésére, lehetővé téve a dinamikus tartalom megjelenítését a weboldalon.

### Git/Github



Git-et verziókezelésre használtam, amely lehetővé tette a változások nyomon követését. GitHub-on tároltam a projekt kódbázisát:  
<https://github.com/KT1M1/Flavora>

# Adatbázis Felépítése

Az adatbázis kialakításánál nagy figyelmet fordítottam a redundancia elkerülésére.

Például egy adott hozzávaló, mint a só, ami számos receptben szerepel, csak egyszer jelenhet meg az adatbázisban, elkerülve, hogy rengeteg receptnél ismételt előforduljon, feleslegesen növelve az adatbázis méretét.

Ehhez kapcsolódó táblákat, mint például a „contains\_ingredient”, használok, amelyek megjelölik, hogy melyik hozzávaló (id alapján) melyik ételben (id alapján) található meg.

Az ismétlődések elkerülése érdekében további stratégiákat is alkalmazok, mint amikor a hozzávalók mértékegységeit előre meghatározott lehetőségek közül lehet kiválasztani, így elkerülve, hogy például a „gramm” vagy a rövidítése „g” többször szerepeljen az adatbázisban.

A flavora nevű adatbázisom ezt a 12 táblát tartalmazza:

food

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(50)	egyedi receptnév
time_min	smallint(6)	hány perc elkészíteni
portion	int	hány adag
description	varchar(300)	rövid leírás
upload_date	date	feltöltés dátuma
user_id	int	user tábla idegen kulcsa

user

Név	Típus	Leírás
id	int	elsődleges kulcs
full_name	varchar(50)	teljes név
user_name	varchar(50)	egyedi felhasználónév
password	varchar(100)	PASSWORD_DEFAULT
email	varchar(100)	egyedi email cím
profile_pic	varchar(100)	regisztráláskor választható
registration_date	timestamp	CURRENT_TIMESTAMP

heart

Név	Típus	Leírás
food_id	int	food tábla idegen kulcsa
user_id	int	user tábla idegen kulcsa

step

Név	Típus	Leírás
id	int	elsődleges kulcs
description	text	recept elkészítésének lépései
food_id	int	food tábla idegen kulcsa

category\_type

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(25)	kategória típusok

category

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(50)	egyedi kategória név
parent_id	int	belső kapcsolat
category_type_id	int	idegen kulcs

	id	name	id	name	parent_id	category_type_id
 Törlés	5	Alkalom	4	főétel	NULL	1
 Törlés	2	Alkategória	5	köztes étkezés	NULL	1
 Törlés	1	Főkategória	6	sült húsok	4	2
 Törlés	3	Konyha	7	desszert	NULL	1
 Törlés	4	Nehézség	8	rizs étel	4	2
			9	tészta étel	4	2
			10	grillezett ételek	4	2
			11	szendvicsek	5	2
			12	wrapek	5	2

10. ábra: category\_type & category tábla

contains\_category

Név	Típus	Leírás
food_id	int	food tábla idegen kulcsa
category_id	int	category tábla idegen kulcsa

ingredient

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(50)	egyedi hozzávaló név

unit

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(25)	egyedi mértékegység név

contains\_ingredient

Név	Típus	Leírás
food_id	int	food tábla idegen kulcs
ingredient_id	int	ingredient tábla idegen kulcs
unit_id	int	unit tábla idegen kulcs
quantity	int	hozzávaló mennyisége

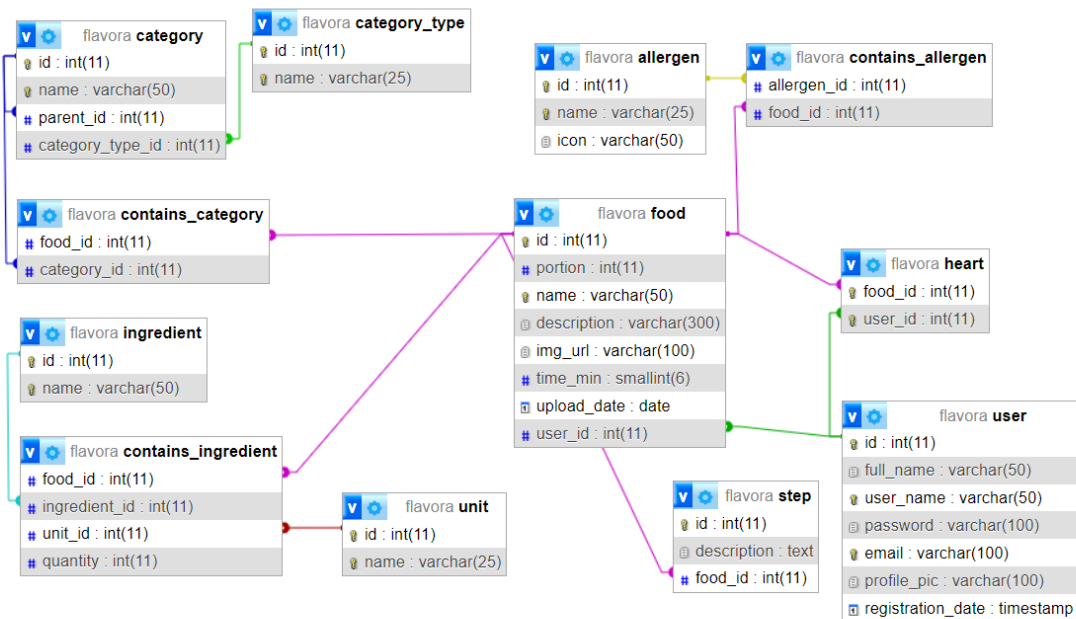
allergen

Név	Típus	Leírás
id	int	elsődleges kulcs
name	varchar(25)	egyedi allergén név
icon	varchar(50)	allergénhez tartozó kép neve

contains\_allergen

Név	Típus	Leírás
allergen_id	int	allergen tábla idegen kulcsa
food_id	int	food tábla idegen kulcsa

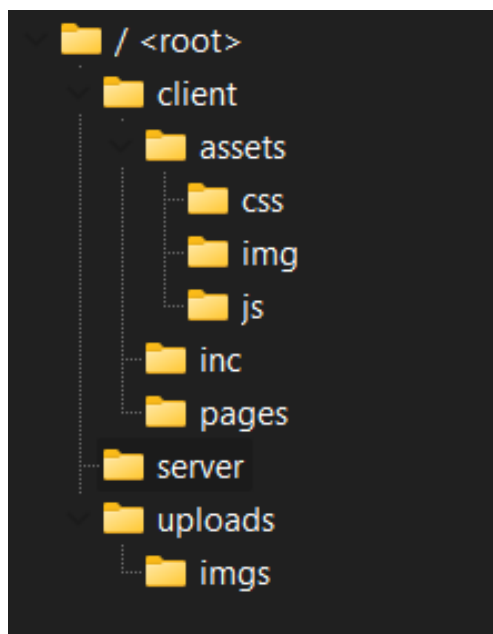
## Adatbázis Kapcsolati Diagram



11. ábra: Adatbázis kapcsolati diagram

A kapcsolati diagram elsődleges célja, hogy ábrázolja a táblák közötti kapcsolatokat. Ez a kép hatékonyan mutatja be ezeket a kapcsolatokat.

# Fájlrendszer



12. ábra: Fájlrendszer

Az „inc” mappában helyezkedik el a header és a footer.

A weboldal dinamikus tartalmát a „pages” mappában található különféle oldalak alkotják. Az index.php, amely a gyökérkönyvtárban található, felelős azért, hogy a header és a footer közötti területen változó tartalmat jelenítsen meg a „pages” mappából választva. Az oldalak a „pages” mappában rendelkezhetnek kiegészítő CSS és JS fájlokkal, amelyeket az erre alkalmas „assets” mappán belüli mappában tárolunk.

A statikus képek, mint például a banner kép, az „img” mappában kapnak helyet. Ezek a képek általában állandó elemek a weboldalon.

Az „uploads/imgs” mappa tartalmazza a felhasználók által feltöltött képeket, például a receptekhez kapcsolódó fotókat.

A „server” mappában találhatóak a szerveroldali PHP kódok, amelyek pl. az adatbázis csatlakoztatásához, adatok lekéréséhez kapcsolódnak.

A kliens oldal az a rész, amit a felhasználók közvetlenül látnak és használnak; ez magában foglalja a weboldalak megjelenését és azok interaktív elemekkel való bővítését.

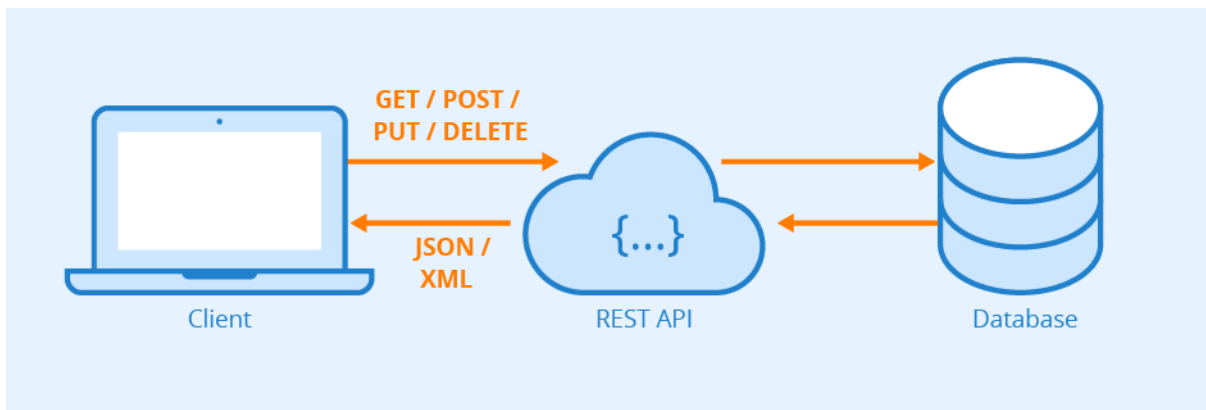
Ezzel szemben a szerver oldal a háttérben dolgozik, adatokat kezel és feldolgoz, valamint gondoskodik a weboldal működéséhez szükséges logika végrehajtásáról. Mindkét oldal elengedhetetlen a modern webes élmény szempontjából, mivel együtt biztosítják, hogy a weboldalak dinamikusak, interaktívak és személyre szabottak legyenek.



# Programozási Technikák

## RestAPI szoftverarchitektúra

A REST API (Representational State Transfer Application Programming Interface) egy szabványos módja az adatok és funkciók hozzáférésének és kezelésének webes alkalmazások vagy szolgáltatások között. Egy REST API lehetővé teszi, hogy különböző szoftveralkalmazások kommunikáljanak egymással HTTP kérések és válaszok segítségével



13. ábra: RestAPI szoftverarchitektúra

## API

A like funkció megvalósítását a weboldalamon egy egyszerű, de hatékony API használatával oldottam meg. Ez a funkció szemlélteti, hogyan alkalmaztam ezt a technikát a projectem során.

```
function set_like(food_id, user_id) {
  var emptyHeart = document.getElementById('heart');

  console.log("Like Set")

  // Send a request to the server to set a like
  fetch(`/server/like.php?food_id=${food_id}&user_id=${user_id}`)
    .then(response => response.json())
    .then(data => {
      console.log( data );
      // Increment the like count displayed next to the heart icon
      emptyHeart.nextElementSibling.innerHTML = (emptyHeart.nextElementSibling.innerHTML * 1) + 1;
    })
    .catch(error => {
      console.error('There has been a problem with your fetch operation:', error);
    });
}
```

14. ábra: Részlet a show.js-ből

Az első lépésben javascript függvény felelős azért, hogy kezelje a felhasználói interakciót. Amikor egy felhasználó rákattint a szív ikonra, ez a függvény aktiválódik. A funkció kapja az étel azonosítóját (food\_id) és a felhasználó azonosítóját (user\_id), amiket paraméterként továbbít egy HTTP GET kérésen keresztül az API-nak.

Ezt követően a JavaScript kód egy fetch kérést indít a /server/like.php címre, paraméterként átadva az étel és felhasználó azonosítóit. A kérés végrehajtása után egy JSON választ kapunk vissza.

A szerver oldalon a „like.php” fájl képezi az API részét, amely kezeli a beérkező kéréseket. Ellenőrzi, hogy a szükséges GET paraméterek (food\_id és user\_id) megtalálhatóak-e. Amennyiben igen, meghívja a set\_like funkciót, ami beszúrja az adatbázisba az új kedvelést. Sikeres művelet esetén egy JSON-t kapunk válaszul, ami jelzi a művelet sikerességét.

Visszatérve a JavaScript kódhoz, ha a válasz sikeres, a függvény frissíti a felületet: növeli a kedvelések számát, ezzel jelzi a felhasználó számára, hogy a kedvelése rögzítve lett. A felhasználói felület frissítése azonnal megtörténik, így a felhasználói élmény gyors és zökkenőmentes.

```
header('Content-Type: application/json');
require_once(__DIR__ . "/show.php");

if(isset($_GET['food_id']) && isset($_GET['user_id'])) {

    set_like( $_GET['food_id'], $_GET['user_id'] );

    echo json_encode((object)[
        "result" => "success"
    ]);
}
```

15. ábra: like.php

```

try {
    $password_hash = password_hash($password, PASSWORD_DEFAULT);

    $sql="INSERT INTO `user` (`full_name`, `user_name`, `password`, `email`, `profile_pic`)
VALUES (:full_name, :user_name, :password, :email, :profile_pic)";
    $stmt = $db->prepare($sql);
    $stmt->bindParam(':full_name', $full_name, PDO::PARAM_STR);
    $stmt->bindParam(':user_name', $user_name, PDO::PARAM_STR);
    $stmt->bindParam(':password', $password_hash, PDO::PARAM_STR);
    $stmt->bindParam(':email', $email, PDO::PARAM_STR);
    $stmt->bindParam(':profile_pic', $profile_pic, PDO::PARAM_STR);

    $stmt->execute();

    login_user($user_name, $password);

    return "sikeres";
} catch (PDOException $e) {
    return "sikertelen " . $e->getMessage();
}
}

```

16. ábra: szerver oldali kódrészlet

## PDO

A PDO (PHP Data Objects) egy adatbázis absztrakciós réteg, ami egységes interfészt biztosít különböző adatbázis-rendszerekhez a PHP alkalmazásokban, vagyis adatbázis független.

### Hibakezelés

Amikor a PDO műveletek során kivétel (exception) keletkezik (például adatbázis kapcsolati hiba), a kód elkapja ezt a PDOException típusú kivételt, és lehetőséget biztosít a hiba kezelésére vagy naplózására. Ez segít az alkalmazás stabil működésében, még hiba esetén is, és lehetővé teszi a fejlesztő számára, hogy tájékozódjon a probléma forrásáról.

### Preparált Lekérdezések és Paraméterek Kötése

Az adatok beszúrása előkészített utasítások „prepare()” segítségével történik, amelyek csökkentik az SQL injekció kockázatát, egy gyakori webbiztonsági fenyegetést. Az előkészített utasítások paraméterezése „bindParam()” lehetővé teszi a változók biztonságos beillesztését a lekérdezésbe. Ezzel a módszerrel meghatározhatjuk az adatok típusát (pl. PDO::PARAM\_STR a szöveges adatokhoz), ami tovább növeli a típusbiztonságot és csökkenti a hibák valószínűségét.

## Hash

Jelszókezelésre a `password_hash` és `password_verify` függvényeket használtam, amelyek biztosítják a felhasználói jelszavak biztonságos tárolását és ellenőrzését.

A titkosításra a `PASSWORD_DEFAULT` opciót használok, ami azt jelenti, hogy a PHP aktuális, ajánlott titkosítási algoritmusát használja, amely idővel frissülhet, hogy a legújabb biztonsági standardoknak megfeleljen. Ez a gyakorlat megnehezíti a rosszindulatú személyek számára, hogy visszafejtsék a jelszavakat, még abban az esetben is, ha hozzáférnek az adatbázis tartalmához.

## Munkamenet Kezelése

A felhasználók állapotának nyomon követésére PHP munkameneteket használtam. Ez lehetővé tette számomra, hogy biztonságosan kezeljem a bejelentkezett felhasználók adatait, és korlátozzam a hozzáférést bizonyos oldalakhoz csak bejelentkezett felhasználók számára.

```
<?php
    session_start();
    require_once("../server/user.php");

    $load_page = "main";

    if(isset($_GET["page"])) {
        $load_page = $_GET["page"];
    }

    $NEED_LOGIN = array(
        "upload_recipe",
        "profile"
    );

    if( in_array($load_page, $NEED_LOGIN) ) {
        validate_user_session();
    }
?>
```

17. ábra: `index.php` első sorai

A recept feltöltése, és a saját profil megtekintése bejelentkezéshez kötött, ezeket a „NEED\_LOGIN” tömbben tárolom, így ez bármikor bővíthető. Ha ezen oldalak valamelyikére akarunk eljutni, meghívódik a szerver oldali „validate\_user\_session” függvény.

```
function login_user($user_name, $password) {
    $user = get_user_by_user_name( $user_name );
    $user = ($user != NULL ? $user[ 0 ] : NULL);

    if($user == NULL || !password_verify($password, $user["password"])) {
        return "Hibás felhasználónév vagy jelszó!\n";
    }

    session_start();
    // Store user data
    $_SESSION['user'] = $user;

    // Redirect to main page
    header('Location: /');
    exit();
}

function logout_user() {
    session_destroy();
    header('Location: /login');
    exit();
}

function validate_user_session() {
    if ( session_status() !== PHP_SESSION_ACTIVE ) {
        session_start();
    }

    if ( !isset( $_SESSION['user'] ) ) {
        header('Location: /login');
        exit();
    }

    return TRUE;
}
```

18. ábra: Felhasználói autentikáció részlet

Az első függvény megkeresi a „user” adatbázisban a megadott felhasználónevet. Ha a felhasználó létezik, és a megadott jelszó megfelel a tárolt jelszónak (a password\_verify() segítségével ellenőrizve), akkor a felhasználói adatokat eltárolja a munkamenetben.

Ezután átirányítja a felhasználót a főoldalra. Ha a felhasználónév vagy jelszó hibás, egy hibaüzenetet ad vissza.

A logout\_user függvény a felhasználó kijelentkeztetéséért felel. Megszünteti a munkamenetet a session\_destroy() függvény hívásával, ami törli az összes munkamenetben tárolt adatot. Ezt követően átirányítja a felhasználót a bejelentkezési oldalra.

A validate\_user\_session függvény ellenőrzi, hogy a felhasználó be van-e jelentkezve. Ha a PHP munkamenet nem aktív, akkor elindítja azt (session\_start()). Ezután megvizsgálja, hogy a munkamenetben van-e eltárolva user adat. Ha nincs, akkor feltételezi, hogy a felhasználó nincs bejelentkezve, és átirányítja a bejelentkezési oldalra. Ha a felhasználó be van jelentkezve (azaz a user adat létezik a munkamenetben), a függvény igaz értékkel (TRUE) tér vissza, jelezve, hogy a felhasználói munkamenet érvényes.

## Dinamikus Tartalomgenerálás

Ez a technika lehetővé teszi, hogy PHP segítségével, az adatbázisból lekért adatok alapján dinamikusan hozzam létre a tartalmat a kliens oldal számára. Az így generált tartalom aztán beilleszthető a HTML oldalba, így a felhasználóknak mindig a legfrissebb, adatbázisból származó információ jelenik meg.

Például a saját profil tartalmának megjelenítésekor így alkalmazom ezt a tudást:

A profiloldal betöltésekor először kinyerjük a felhasználó adatait a sessionből:

```
$user = $_SESSION["user"];
```

Ezután meghívjuk a szerver oldali funkciót, amely visszaadja a felhasználó által kedvelt recepteket:

```
$liked_recipes = get_liked_recipes($user["id"]);
```

```
function get_liked_recipes($user_id) {  
    global $db;  
  
    $stmt = $db->prepare(  
        'SELECT food.id, food.name, food.img_url  
        FROM heart  
        JOIN food ON heart.food_id = food.id  
        WHERE heart.user_id = :user_id'  
    );  
  
    $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);  
    $stmt->execute();  
  
    $result = $stmt->fetchAll(PDO::FETCH_ASSOC);  
  
    return $result;  
}
```

19. ábra: Kedvelt receptek lekérése

```

<div class="row">
  <?php
  foreach($liked_recipes as $recipe) {
    echo '
    <div class="col-12 col-sm-6 col-lg-4">
      <a href="show_recipe/' . $recipe['id'] . '" class="card-div-link">
        <div class="card-div">
          <div class="card-top">
            
          </div>
          <div class="card-bottom">
            <div class="card-text">
              <p>' . $recipe["name"] . '</p>
            </div>
          </div>
        </div>
      </a>
    </div>
    ';
  }
  ?>
</div>

```

20. ábra: Dinamikus tartalomgenerálás

Miután megkaptuk a lekérdezés eredményét, a PHP foreach ciklusát használjuk, hogy végigmenjünk az összes rekordon. Minden iteráció során dinamikusan generálunk egy HTML „card” elemet, amely tartalmazza a recept nevét, képét, és egy linket a részletes oldalára. Ez a dinamikus generálási folyamat lehetővé teszi, hogy a weboldal valós időben frissüljön új adatokkal anélkül, hogy statikusan kellene frissíteni az oldalt minden egyes adatváltozás esetén.



## Weboldal Optimalizálása

A weboldal optimalizálása magában foglalja a weboldal technikai és vizuális elemeinek fejlesztését. Ez tartalmazza a betöltési idő csökkentését, a felhasználói felület átgondolt kialakítását. Egy jól optimalizált weboldal nemcsak, hogy vonzóbbá válik a látogatók számára, de használata is egyszerűbbé és intuitívabbá válik.

A gyorsaság érdekében a weboldalon lévő állandó képeket webP formátumba konvertáltam, ami jelentősen csökkenti a fájl méreteket anélkül, hogy észrevehetően romlana a képminőség. Ezenkívül a felhasználók által feltöltött képek méretét is korlátozom, hogy elkerüljem a nagyméretű fájlok lassú betöltését, ami negatívan befolyásolhatja az oldal teljesítményét és a felhasználói élményt.

## Keresőmotor optimalizálása

A keresőmotor optimalizálás (SEO) egy másik kritikus aspektusa, amely a weboldal tartalmának és szerkezetének módosítását jelenti úgy, hogy az jobban teljesítsen a keresőmotorokban, mint például a Google. Ez magába foglalja a kulcsszavak megfelelő használatát, meta címkék (például a cím, leírás és kulcsszavak) beállítását.

```
<!DOCTYPE html>
<html lang="hu">
<head>
  <title>Flavora</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Böngéssz gyorsan és egyszerűen mások által feltöltött recept között, vagy töltsd fel sajátod ezen a mag
  <meta name="keywords" content="receptmegosztó, magyar receptek, recept feltöltés, online receptkönyv, receptkereső, konyhaművészet, egysz
```

21. ábra: Főoldalam head részlege

`<meta name="description">`: Ez a leírás tag, ami rövid összefoglalót ad a weboldal tartalmáról.

`<meta name="keywords">`: Kulcsszavakat tartalmaz, ami segíti a weboldalt abban, hogy releváns keresési kifejezésekhez kapcsolódjon.

A szemantikus HTML elemek használata, mint `<header>`, `<footer>`, `<nav>`, segíthet a tartalom jobban strukturált és értelmezhetővé tételében mind a felhasználók, mind a keresőmotorok számára.

A weboldalon használt összes képhez az `<alt>` címkék hozzáadása növeli az oldal hozzáférhetőségét, mivel a látássérültek számára használt képernyőolvasó szoftverek ezek alapján képesek leírni a képek tartalmát. Emellett, ha egy kép valamilyen okból nem töltődik be, az `<alt>` szöveg jelenik meg helyette.

## További Fejlesztési Tervek

Tervezek néhány lehetséges fejlesztést a weboldalam számára a jövőben, amelyekkel tovább gazdagíthatnám a felhasználói élményt. Ezek a funkciók nem elengedhetetlenek az oldal működéséhez, de hozzáadásukkal növelhetném a felhasználók elégedettségét.

### Elfelejtett Jelszó

Az elfelejtett jelszó funkció lehetővé tenné a felhasználók számára, hogy egyszerűen és biztonságosan állíthassanak be új jelszót, amennyiben az eredetit elfelejtették. A helyreállítás e-mail cím megadásával lenne lehetséges, ahová egy jelszó-visszaállító linket küldenénk.

### Recept Módosítása Feltöltés Után

Jelenleg a felhasználók képesek receptjeiket törölni esetleges hiba esetén, azonban nincs lehetőségük a már feltöltött receptek módosítására. Ez a funkció hasznos lenne azok számára, akik véletlenül hibás információt adtak meg a receptjükben. Jelenleg egy kis elírás sok bajlódással járna, amit ez a funkció megoldhatna.

### Videó Feltöltési Lehetőség

A jelenlegi állapotban a felhasználók csak képeket tölthetnek fel receptjeik mellé. A jövőben szeretném lehetővé tenni videók feltöltését is, ami még gazdagabbá és érthetőbbé tehetné a receptek bemutatását.

# Források

A weboldalam megalkotása során számos külső erőforrásra támaszkodtam, amelyek nélkülözhetetlenek voltak a végső kivitel megvalósításához. Az alábbiakban szeretnék megköszönni minden egyes szolgáltatásnak és technológiának, amelyek hozzájárultak projektem sikeréhez.

## Képszerkesztés:

A profilképek egyedi megalkotásához először a Bing AI képgeneráló szolgáltatását vettem igénybe. Az így létrejött alap képeket aztán az Adobe Photoshop segítségével finomítottam és személyre szabtam, hogy minden egyes kép tökéletesen illeszkedjen weboldalam stílusához.

A használt képek átalakításához korszerű és optimalizált formátumra, valamint a méretük csökkentéséhez a Gimp programot használtam.

## Képek a weboldalhoz:

Weboldalam látványvilágának színesítése érdekében számos megbízható forrásból válogattam ingyenesen felhasználható képeket:

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

<https://pixabay.com/>

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

<https://unsplash.com/>

## Ikonok:

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

## Weboldalon használt betűtípus:

<https://fonts.google.com/>

## Weboldal publikálása:

A weboldalam hostingját a Nethely szolgáltatása tette lehetővé.