

BKSZC Weiss Manfréd Technikum,
Szakképző Iskola és Kollégium

VIZSGAREMEK

Témavezető:
Kovács László

Készítette:
Csonka Dominik,
Mohai Olivér

BUDAPEST
2025

Tartalomjegyzék

Bevezetés.....	5
Témaválasztás indoklása	5
Célkitűzés.....	5
Kiknek szánjuk a weboldalt	5
Fejlesztői dokumentáció	6
Fejlesztői környezet	6
Domain és tárhely	6
Miért InfinityFree?.....	6
Fejlesztői dokumentáció: Közösségi média.....	7
Fejlesztéshez használt hardver	7
Tesztelés és tesztdokumentáció.....	7
Tesztelési esetek és visszajelzés	7
Tesztbejelentkezési adatok.....	7
Adatbázis táblák.....	8
' <i>user</i> ' tábla (<i>felhasznált részei</i>).....	8
' <i>dok</i> ' tábla	9
' <i>votelog</i> ' tábla	9
Algoritmusok a weblapon.....	10
Jogosultságellenőrzés	10
Fájlkezelés	11
Szavazási opciók kezelése	12
Bejegyzések megkülönböztetése.....	13
Weboldal dizájnja	13
Különböző körülmények, esetek és hibakezelések.....	14
Interakció következtében megjelenő hibakezelés	14
Háttérben folyó hibakezelés	14
Fejlesztési lehetőségek.....	15
Fejlesztőidokumentáció: Csevegő felület	16
Különbségek már létező programoktól	16
Fejlesztői környezet	16
Felhasznált nyelvek	16

Kiegészítő elemek.....	17
A fejlesztéshez használt hardver	17
Adatszerkezet	18
“friendship” tábla	19
“uzenetek” tábla	20
Algoritmusok.....	21
Csevegések megjelenítése:	21
Barát hozzáadása	23
Üzenet küldése	24
Fejlődési lehetőség.....	25
Merre tovább?	25
Működési alapelvek	25
Csoportok kialakítása.....	25
Teljes modernizálás.....	25
Tesztelői hozzáférés	26
Felhasználói dokumentáció.....	27
A weboldal célja	27
Program használatához szükséges eszköz	27
Program elindítása	28
Egyszerű keresés:	28
Keresés URL alapján:	28
Az oldalra érkezés után:	29
Bejelentkezés után:	29
Felhasználói dokumentáció: Közösségi média	30
Az oldal elérése	30
Általános jogosultságok	31
Kereső funkció	31
Hozzászólás írása	31
Szavazás lehetősége	32
Saját hozzászólások törlése	32
Letöltés	33
Diákönkormányzatos jogosultságok.....	34

Új bejegyzések létrehozása	34
Bejegyzések és hozzászólások kezelése	37
Felhasználói dokumentáció: Csevegő felület	38
Program használata.....	38
A barát hozzáadás után:.....	40
Összefoglalás	42
Értékelés:.....	42
Program hasznosulása	42
Köszönetnyilvánítás.....	43
Irodalomjegyzék	44

Bevezetés

Témaválasztás indoklása

Mindenképp szerettünk volna egy olyan weboldalt létrehozni, amely egy valós problémára ad gyakorlati megoldást. Emellett fontos szempont volt, hogy a projekt kihívás elé állítson bennünket, és lehetőséget adjon a már meglévő ismereteink fejlesztésére. Kezdetben többféle elképzelés is felmerült, azonban ezek többsége nem igényelt volna többfős együttműködést, így a projekt témájának kiválasztása nehézséget okozott. Tapasztalataink szerint, az iskolánk tanulói nem mindig kapták meg időben a diákönkormányzati gyűlésen elhangzott információkat. Ez alapján kaptunk egy olyan témát, amely ennek a kommunikációs problémának az áthidalását célozza. Bár kezdetben attól tartottunk, hogy ebben a projektben is nehéz lenne a feladatokat úgy megosztani, hogy közben ne zavarjuk egymás munkáját, végül sikerült egy olyan ötletet találni, amely ezt áthidalja. Az eredeti ötletet továbbgondolva, a felmerült igények figyelembevételével született meg a gólyakereső, valamint a privát és csoportos csevegés ötlete.

Célkitűzés

A cél egy fórumsszerű oldal, valamint egy csevegő felület létrehozása volt, amely az intézmény diákjai között hoz létre egy kétrányú kommunikáció lehetőséget, legyen az egyik fél diákönkormányzati tag, vagy egy szimpla diák. A *közösségi média* és csevegő felület ugyan nagy részben független egymástól, de mégis találtunk megoldást arra, hogy egybekössük a kettőt a következőképp:

- A felhasználók nevére vagy profilképére kattintva az oldal gyors elérést biztosít a két felhasználó közti csevegési felülethez.
- A diákönkormányzati tagok által közzétett bejegyzésekben lehetőség van osztályok megjelölésére. Ezek a megjelölések a csevegőfelületen „rendszerüzenetként” jelennek meg, jelezve az érintett diákok számára, hogy osztályuk említve lett egy bejegyzésben.

Kiknek szánjuk a weboldalt

A weboldalt elsősorban a Weiss Manfréd Technikum diákjainak szánjuk – legyenek azok frissen belépő „gólyák”, jelenlegi tanulók, vagy már végzett diákok. A cél, hogy az iskolai közösség tagjai könnyebben elérjék az információkat és kapcsolatba léphessenek egymással.

Fejlesztői dokumentáció

Fejlesztői környezet

A weboldal fejlesztése során több különböző eszközt és környezetet alkalmaztunk, amelyek segítették a munka hatékonyságát és átláthatóságát.

A **Visual Studio Code** egy könnyen kezelhető, széles körben használt kód szerkesztő, amely lehetővé tette a HTML, CSS, JavaScript és PHP kód gyors és hatékony írását.

A **XAMPP** egy olyan komplett fejlesztői csomag, amely tartalmazza az Apache webszervert, a PHP értelmezőt, valamint a MariaDB-t (MySQL-kompatibilis adatbáziskezelő). Ennek segítségével helyileg is futtathattuk és tesztelhettük weboldalunkat.

A **phpMyAdmin** segítségével könnyedén kezelhetjük és karbantarthatjuk az adatbázisunkat, illetve webalapú felületet biztosít a MySQL/MariaDB adatbázisok kezelésére.

A **MySQL** strukturált adatbázis-kezelő rendszer, amely biztosítja az adatok tárolását, visszakeresését és manipulációját.

Ezeket a fejlesztőieszközöket választottuk, mivel korábbi tanulmányaink során már megismertük őket, széles körben elterjedtek, ingyenesen elérhetők, és hatékonyan támogatják a weboldal fejlesztését és tesztelését. A projekt előrehaladtával kényelmi szempontból áttértünk egy online fájlkezelő szerkesztőre is, amely lehetővé tette, hogy bármilyen eszkösről, külön letöltés nélkül elérjük és módosítsuk a forrásfájlokat.

Domain és tárhely

Ahhoz, hogy a weboldal online is elérhető legyen, szükség volt egy domain név és egy megfelelő tárhely kiválasztására. Választásunk az [InfinityFree](#)-re esett, mert a legtöbb igényünket ki tudta elégíteni. Az egyetlen dolog, ami nem volt benne az ingyenes csomagba, az az email kiküldés értesítés, vagy megerősítés szempontjából.

Ettől eltekintve végül itt hoztuk létre a [weissesvagyok.info.uk](#) weboldalt, amely az általunk véglegesített verziója a [weissesvagyok.hu](#) oldalon is elérhető (lesz).

Miért InfinityFree?

Fejlesztésünk első szakaszában a Nethely szolgáltatását próbáltuk ki. Azonban már az induláskor adatbázis-kezelési hibákba ütközünk, és ezek elhárítására nem találtunk működő megoldást. Emiatt praktikusabbnak bizonyult új szolgáltató után nézni, és így esett a választás az InfinityFree-re, amely azóta is stabil működést biztosít.

Fejlesztői dokumentáció: Közösségi média

Fejlesztéshez használt hardver

- Processzor: AMD Ryzen 5 7600X3D
- Videókártya: ASUS ROG STRIX GeForce RTX 4070 SUPER
- Memória mérete: 32 GB
- Operációs rendszer: Windows 11, 64 bit

Tesztelés és tesztdokumentáció

- Böngésző: OperaGX, Google Chrome, Microsoft Edge
- Felbontás: FullHD (1920×1080)
- Mobil eszközök: Android (Chrome), iOS (Safari)
- Chrome DevTools reszponzív módja

Tesztelési esetek és visszajelzés

- 1) Esemény: Fájl feltöltés túl nagy fájllal (>2MB)
Hibaüzenet: „A(z) x. fájl mérete túl nagy, kérem távolítsa el a listából!”
- 2) Esemény: Közzététel üres szöveges beviteli mezővel
Hibaüzenet: „A közzétételhez írjon valamit a mezőbe!”
- 3) Esemény: 1db szavazati opció megadása
Hibaüzenet: „Legalább 2 opciót adjon meg a szavazás létrehozásához!”
- 4) Esemény: Eseményként kezelés beállítása dátum nélkül
Hibaüzenet: „Adjon meg egy lejáratú dátumot!”
- 5) Esemény: Oldal meglátogatása osztály nélkül
Hibaüzenet: „Állítson be osztályt adatlapján!”

Tesztbejelentkezási adatok

Mint általános felhasználó:

Felhasználónév: teszt

Jelszó: teszt123

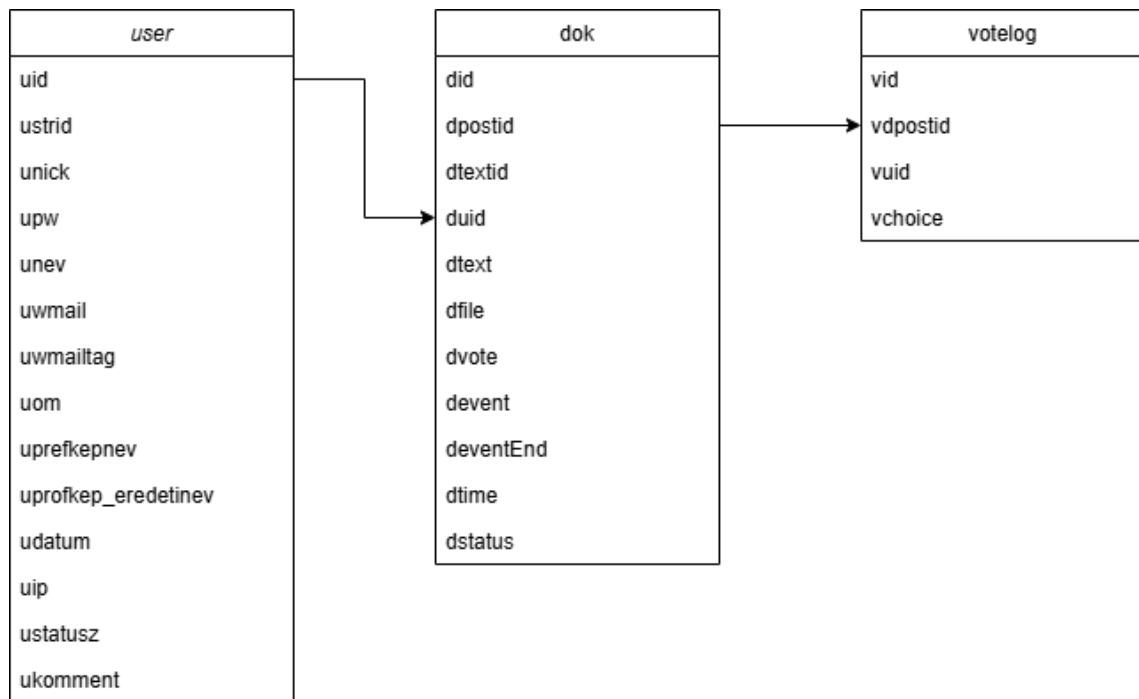
Mint diáknakormányzati tag:

Felhasználónév: tesztdok

Jelszó: teszt123

Adatbázis táblák

A weboldal közösségi funkcióinak alapját képezik az adatbázistáblák, amelyek a felhasználók, bejegyzések, szavazások és egyéb kapcsolódó adatok strukturált tárolását teszik lehetővé. Az adatok elrendezése és a mezők kialakítása során fontos szempont volt az átláthatóság, a kompaktság, valamint az, hogy későbbi bővítés esetén is egyszerűen módosíthatók legyenek az összefüggések.



'user' tábla (felhasznált részei)

- uid – Ez a tábla a felhasználó azonosítója, és a tábla elsődleges kulcsa.
- unick – Ez tartalmazza a felhasználó nevét, amelyet módosíthat is. minden felhasználó neve legfeljebb 16 karakter hosszú lehet.
- uprofkepnev – A felhasználó profilképének egyedileg elmentett neve. A típusa VARCHAR.
- ustatusz – Ez alapján határozható meg a felhasználó jogosultsága.

uid	unick	uprofkepnev	ustatusz
1	Z4tonyy	1_250413020605.jpg	D
2	bot	2_250317055516.png	D
3	dad	3_250411005444.jpg	D
4	nineszko345	4_250330141652.jpg	A
5	Vivien		A
6	sfde		A

'dok' tábla

- did – Ez a tábla elsődleges kulcsa.
dok
did
- dpostid – Ez a közzétett bejegyzés azonosítója.
dpostid
- dtexid – Ez a bejegyzésekhez tartozó komment azonosítója. A 0. azonosítójú „komment” az a bejegyzés önmaga.
dtexid
- dtext – Ez az adatbázisba feltöltött szöveg.
dtext
- dfile – Ez a feltöltött fájlnevek mezője, több esetén ; karakterrel elválasztva.
dfile
- dvote – Ezek a szavazati opciók, több esetén ; karakterrel elválasztva
dvote
devent
- devent – Ez az eldöntés, hogy kiírásnál feltüntesse-e az események között: Igen („I”) / Nem („N”)
deventEnd
dtime
- deventEnd – Ez az esemény lejáratú dátuma. „N” devent tulajdonság esetén az értéke NULL.
dstatus
- dtime – Ez a közzététel ideje.
- dstatus – Ez a bejegyzés állapota Aktív („A”) / Inaktív („I”)

'votelog' tábla

- vid – Ez a tábla elsődleges kulcsa.
votelog
vid
- vdpostid – Ez annak a bejegyzésnek az azonosítója, amelynél a felhasználó éppen szavazott.
vdpostid
vuid
- vuid – Ez a szavazott felhasználó azonosítója.
vchoice
- vchoice – Ez a választott szavazat, szövegként elmentve.

Algoritmusok a weblapon

A közösségi felület működése során több kulcsfontosságú algoritmus is fut a háttérben. Ezek célja az adatok validálása, a hibák megelőzése, valamint a funkciók gördülékeny működtetése.

Jogosultságellenőrzés

A weboldal biztonsága és személyre szabott működése érdekében minden oldalbetöltés előtt ellenőrizzük a felhasználó jogosultságait:

Bejelentkezés ellenőrzése: Ha a felhasználó nem jelentkezett be, átirányítjuk a bejelentkező oldalra.

```
<?php  
$m1 = $_GET['m1'];  
  
if (!isset($_SESSION['uid'])){  
    include("bekellepni.php");  
    die();  
}
```

Osztály beállításának ellenőrzése: Ha a classid mező nincs kitöltve, a rendszer a profiloldalra navigál.

DÖK jogosultság ellenőrzése: A ustatusz = 'D' mező alapján korlátozott funkciókhöz (pl. bejegyzés létrehozás, szavazás indítása) csak DÖK tag férhet hozzá.

```
else if ($user['ustatusz']=='D'){  
    print('  
        <form action="dok_post_ir.php" method="post" enctype="multipart/form-data" target="kisablak" style="background: transparent; box-shadow: none;">  
            <dialog id="voteWindow">  
                <div>  
                    <h2>Opciók megadása:</h2>  
                    <p id="votes-container">  
                        <input type="text" name="input0" maxlength=26 onfocus="addVote.call(this)" onclick="addVote.call(this)"><br>  
                    </p>  
                    <br><br>  
                    <p style ="font-size: 11px;"><u>Megjegyzés:</u><br>  
                    <i>Abban az esetben, ha nem szeretné szavazást létrehozni, vagy mégsemcs szüksége egyes mezőkre, hagyja öket üresen!</i></p>  
                    <label id="closeVote">Kész</label>  
                </div>  
            </dialog>  
        <textarea name="dtext" onclick="this.style.height='215px'"></textarea>  
  
        <label for="dok-post-vote"" onclick="Vote()">Szavazás létrehozása</label><input type="button" name="dok-post-vote" style="display: none">  
  
        <label for="post-file">Fájl csatolása</label><input type="file" name="dok-post-files[]" id="post-file" multiple style="display: none">  
        <ol id="file-list"></ol>;  
  
        print('  
            <dl>  
                <dt>Szeretné eseményként kezelni? </dt>  
                <dd style="margin-left: 13px"> Igen <input type="radio" id="dok-event-add" name="add-event" value="igen"> </dd>  
                <dd style="margin-left: 130px" id="event-date" hidden> Dátum: <input type="date" id="dok-event-date" name="event-date"></dd>  
                <dd style="margin-left: 130px"> Nem <input type="radio" id="dok-event-add" name="add-event" value="nem" checked> </dd>  
            </dl>  
  
            <label for="dok-post-send">Közzététel</label><input type="submit" id="dok-post-send" style="display: none">  
        <hr>  
    </form>  
');  
}
```

Biztonság: A jogosultságokat minden PHP oldalon, szerveroldalon ellenőrizzük, nem csak JavaScript-tel a kliensoldalon.

Fájlkezelés

A felhasználók képesek fájlokat csatolni a bejegyzésekhez (nem hozzászólásokhoz).

- **Feltöltés:** A fájlokat a dok-files/ mappába mentjük. A fájlnév ütközések elkerülése érdekében egyedi fájlnévet generálunk, és ezt mentjük az adatbázisba. A zárt hozzáférés végett nincs megszabva fájlformátum, minden fájl típus engedélyezve van, de csak maximum csak 2 megabajtos fájlméret (fájlonként).

```
//feltöltött fájl kezelése
$files = [];

if (!empty($_FILES['dok-post-files']['name'][0])) {
    foreach ($_FILES['dok-post-files']['name'] as $fi => $fileName) {
        $fileTmpName = $_FILES['dok-post-files']['tmp_name'][$fi];
        $fileSize = $_FILES['dok-post-files']['size'][$fi];

        //2MB max
        if ($fileSize >= 2000000) die("<script>alert('A(z " . $fi + 1 ."- fájl mérete túl nagy, kérlek távolítsa el a listából! (max 2MB)'</script>");

        //prefix: dok-postid-file
        $filePrefix = "dok-" . $postid . "-" . basename($fileName);
        if (move_uploaded_file($fileTmpName, 'dok-files/' . $filePrefix)) $files[] = $filePrefix;
    }
}
$dfile = implode(';', $files);
```

- **Letöltés:** A fájlokat URL-en keresztül lehet letölteni, egy PHP script közvetíti a lekérést. Ez biztosítja, hogy csak létező és jogosult fájlokat lehessen elérni.

```
<?php
    session_start();
    include('kapcsolat.php');
    $m1 = $_GET['m1'];

    if (!isset($_SESSION['uid'])){
        print("<script> alert('Szép próbálkozás! :))</script>");
        die();
    }

    $file = basename($_GET['file']);
    $filePath = 'dok-files/' . $file;

    if (file_exists($filePath)) {

        // ^ = Matches the beginning of the filename (not something in the middle)
        // dok- = Matches the literal text "dok-"
        // \d+ = Matches one or more digits (\d means any number, + means one or more)
        // - = Matches the literal hyphen (-) after the number
        // / = Regex delimiters (used to enclose the pattern)
        $noPrefix = preg_replace('/^dok-\d+-/', '', $file);

        header("Content-Description: File Transfer");
        header("Content-Type: application/octet-stream");
        header("Content-Disposition: attachment; filename=\"$noPrefix\"");
        readfile($filePath);
        exit;
    } else {
        die("<script> alert('Hiba: A fájl nem található.'</script>");
```

- Megjelenítés:** A fájl eredeti neve jelenik meg, nem a generált technikai név.

```
//feltöltött fájl kiírás
$files = explode(';', $dok['dfile']);
print("<div class='postedFile'>");
foreach ($files as $file) {
    if (!empty($file)) {

        // ^ = Matches the beginning of the filename (not something in the middle)
        // dok- = Matches the literal text "dok"
        // \d+ = Matches one or more digits (\d means any number, + means one or more)
        // - = Matches the literal hyphen (-) after the number
        // / = Regex delimiters (used to enclose the pattern)
        $noPrefix = preg_replace('/^dok-\d+/', '', $file);

        print('<a href="file-download.php?file='. urlencode($file) ."' target="kisablak">' . htmlspecialchars($noPrefix) .'</a>');
    }
}
```

Szavazási opciók kezelése

A DÖK tagok bejegyzéshez szavazást is mellékelhetnek.

- Adatrögzítés:** A dvote mező tartalmazza az opciókat ; karakterrel elválasztva.

```
//szavazati opciók
$inputCount = 0;
$dvote = "";
while(isset($_POST['input'][$inputCount])){
    $inputText = trim(preg_replace('/[ ]{2,}/', ' ', $_POST['input'][$inputCount]));

    if ($inputText!="") $dvote = $dvote . $inputText . ";";
    $inputCount++;
}
$dvote = substr($dvote, 0, -1) . ' ';
```

- Megjelenítés:** A szavazási opciókat explode(';) függvénytel tömbbe alakítjuk, majd a tömböt végig járva, minden elemet rádiógombként jelenítünk meg.

```
$votes= explode(';', $dok['dvote']);      for($i=0; $i < count($votes); $i++){
```

- Szavazat leadása:** A votelog táblába új rekord kerül, amely tartalmazza a felhasználó azonosítóját, a szavazott bejegyzés dpostid mezőjét, és a választott opció szövegét.

- Újraszavazás:** Az előző szavazat módosítható, az új érték felülírja a régit.

```
<?php
session_start();
include("kapcsolat.php");

$dpostid = intval($_POST['dpostid']);
$vuid = $_SESSION['uid'];
$vchoice = $_POST[$dpostid];
$exists = mysqli_query($adb, "SELECT * FROM votelog WHERE vuid = $vuid AND vdpostid = $dpostid");

if (mysqli_num_rows($exists) == 0) mysqli_query($adb, "
    INSERT INTO votelog (vdpostid, vuid, vchoice)
    VALUES      ('$dpostid', '$vuid', '$vchoice')");
else mysqli_query($adb,
    UPDATE votelog SET vchoice = '$vchoice'
    WHERE vuid = $vuid AND vdpostid = $dpostid");

print("<script>window.parent.location.reload();</script>");
?>
```

Bejegyzések megkülönböztetése

A rendszer képes kezelni a bejegyzések és hozászólások hierarchiáját.

- **Bejegyzés:** dtexid = 0, dpostid egyedi (új bejegyzés létrehozásakor a jelenlegi legnagyobb dpostid + 1).

```
$postid = mysqli_fetch_array(mysqli_query($adb, "SELECT MAX(dpostid) FROM dok"))[0] + 1;
```

- **Komment:** dtexid > 0, dpostid megegyezik a szülőbejegyzésével. A dtexid értékét szintén a jelenlegi legnagyobb érték + 1 alapján határozzuk meg.

```
<?php
    session_start();
    include("kapcsolat.php");

    $textid = mysqli_fetch_array(mysqli_query($adb, "SELECT MAX(dtexid) FROM dok WHERE dpostid = $_POST[dpostid]"))[0] + 1;
    $text = isset($_POST['commentInput']) ? trim($_POST['commentInput']) : '';
    if ($text == '') die();

    mysqli_query( $adb , "
        INSERT INTO dok (
            dpostid , dtexid , duid , dtext , dtime , dstatus
        VALUES
            ( $_POST[dpostid] , $textid , $_SESSION[uid] , '$text' , NOW() , 'A' )
        ");
    print("<script>setTimeout(function() {window.parent.location.reload()}, 100)</script>");
?>
```

- **Összekapcsolás:** A rendszer bármilyen bejegyzés- és kommenttípushat képes kezelni, akár fájlmelléklettel, akár szavazással együtt.

Weboldal dizájnja

A felületet próbáltam a lehető legegyszerűbben megoldani, hogy senkinek se akadjon kétsége, amikor ezt az oldalt használja. Az oldalam frontend részét barátaimra hagyatkozva oldottam meg, hisz én épp az egyszerűség miatt az alap formázással és színezéssel is beértem volna. Az oldal mobilbarát kialakítású. minden dinamikusan jelenik meg, így akár egy kisebb, vagy épp egy nagyobb képernyőn is tökéletesen működik.

A bejegyzések és kommentek elkülönülnek, a profilkép, név, időbélyeg és mellékletek jól strukturáltan jelennek meg. A szavazások és a fájlok külön blokkokban kapnak helyet, megkönnyítve az áttekintést. Az esemény fül 1210 pixelszélességen túl nem elérhető, megjelenítése meggondolandó. Ez a mérték szándékosan megegyezik az index „mobilnézetével”, az egységes kinézet miatt.

Különböző körülmények, esetek és hibakezelések

A közösségi média modul működése során számos olyan helyzettel találkozhatunk, amely rendellenes működéshez, hibákhoz vagy jogosultsági problémákhöz vezethet. Ezek megfelelő kezelése elengedhetetlen a stabilitás, biztonság és felhasználói élmény biztosítása érdekében.

A fejlesztés és tesztelés során kiemelt figyelmet fordítottunk a beviteli hibák, a fájlkezelés során fellépő problémák, a jogosultsági ellenőrzések, a szavazások integritása, valamint az adatbázis-kezelés biztonságos működésének biztosítására.

Interakció következtében megjelenő hibakezelés

Jó pár interakció következtében az oldal frissülni kényszerül. Ennek az az oka, hogy a felhasználó ne csak a saját maga által éppen kitett posztot lássa, hanem esetleg új megjelenésű bejegyzéseket, hozzájárulásokat, vagy akár a szavazók számát. A beviteli hibák kezelése során a rendszer ellenőrzi, hogy a felhasználó nem hagyta-e üresen a kötelező mezőket. Például egy új bejegyzés vagy hozzájárulás beküldése csak akkor engedélyezett, ha a szövegmező nem üres. Ezen kívül (jelenleg csak szavazási opciók megadásánál) bizonyos helyeken meg van határozva maximális karakterkorlát. A szavazások esetében a rendszer megköveteli, hogy legalább két opció kerüljön megadásra, ellenkező esetben figyelmeztető üzenet jelenik meg, és a szavazás mentése meghiúsul.

A fájlkezelésnél hiba esetén a közzététel gomb megnyomása után felszólít, hogy a 2MB-nál nagyobb méretű fájlt, aminek a sorszámát ki is írja a felugró párbeszéd ablakban, hogy távolítsa el a fájl neve mellett megtalálható piros X megnyomásával.

Háttérben folyó hibakezelés

Jogosultsági szempontból a rendszer már az oldal betöltésekor ellenőrzi, hogy a felhasználó be van-e jelentkezve. Amennyiben ez nem teljesül, automatikusan a bejelentkező felületre irányítja a látogatót. A regisztrált felhasználóknak is meg kell felelniük bizonyos feltételeknek: például amennyiben a profilukhoz nincs hozzárendelve osztály. Ilyen esetben a rendszer figyelmeztet, majd a profiloldalra irányítja őket. minden adatbázisból történő kiíratásnál el lett végezve a HTML tagok elleni védelem. Magyarán, ha akármelyik beviteli mezőbe például egy „<button>” szócska kerül, akkor az kiírva nem egy HTML tag-ként, hanem nyers szövegként jelenik meg. A túlzott adatbázis-terhelést megelőzve kompakt adatbázis lett a cél, illetve tényleges naplózás helyett csak UPDATE fut le az adatbázisban, nem pedig CREATE. Az egyedi azonosítók, például a dpostid és dtextrid generálásakor a rendszer a közzététel gomb megnyomására lekérdezi az aktuális legnagyobb értéket, és azt követően határozza meg az új értéket. Ez a megközelítés megbízható módon akadályozza meg az

azonosítóütközéseket, különösen párhuzamos felhasználás esetén, ami a jelenlegi esetben hatalmas befolyásoló tényező.

Fejlesztési lehetőségek

A jelenlegi közösségi média modul számos funkcióval rendelkezik, amelyek lehetővé teszik a felhasználók közötti kommunikációt, a tartalmak megosztását, szavazások lebonyolítását, valamint fájlok kezelését. Ugyanakkor a fejlesztés során számos olyan területet azonosítottunk, ahol a jövőben tovább lehetne bővíteni vagy finomítani a rendszert, a funkcionalitás növelése, a biztonság erősítése és a felhasználói élmény javítása érdekében.

Az egyik legfontosabb fejlesztési irány a valós idejű funkciók bevezetése. Jelenleg a rendszer csak akkor frissíti a tartalmakat, ha a felhasználó manuálisan újratölти az oldalt, vagy bizonyos műveletek automatikusan újratöltést váltanak ki. Ezt kiválthatná egy WebSocket-alapú megoldás, amely lehetővé tenné, hogy az új hozzászólások, szavazatok vagy fájlfeltöltések automatikusan megjelenjenek a felhasználók felületén. Továbbá fontolóra lehetne venni a bejegyzésekhez ugrást csak egy URL ki- és bemásolásával, ezzel is lehetőséget adni egyes bejegyzések továbbítására. Ezekkel az oldal sokkal dinamikusabbá válna, és jobban megfelelne a mai felhasználói elvárásoknak.

Végül, érdemes lenne megvizsgálni egy mobilalkalmazás fejlesztésének lehetőségét is, amely a webes felület funkcionalitását natív formában kínálná Android és iOS eszközökön. Ez nemcsak kényelmesebb elérést biztosítana, hanem lehetőséget nyújtana olyan eszközspecifikus funkciók kihasználására is, mint az értesítések, kamera használata fájlcsatoláshoz vagy offline tartalmak mentése.

Fejlesztőidokumentáció: Csevegő felület

Különbségek már létező programoktól

A Weissesvagyok.hu csevegő felülete elvi működésben nem sokban tér el a különféle már létező élő csevegésre alkalmas programoktól. Leginkább a felhasznált technológiák azon jellemzők melyek elválasztják a többi hasonló programtól. A fejlesztés során a magasabb szintű technológiák helyett, mint például WebSocket a csevegő felülethez folyamatosan újra frissítő JavaScript logikát használtunk mivel magasabb szintű technológiák esetében tudásunk nem érte el a megfelelő szintet a minőségi munka biztosításához. A program emellett különbözik más populáris csevegő felületektől hiszen ezen program a Weissesvagyok.hu oldalán fut biztosítva az ott kialakuló közösséggépítést és könnyű mozgást az oldal többi felületére.

Fejlesztői környezet

A fejlesztés során többféle technológiát alkalmaztunk, elsősorban a XAMPP nevezetű program csomagot. A XAMPP program csomag tartalmazza az Apache helyi webszerver szolgáltatót, a MySQL adatbáziskezelőt és a PHP támogatáshoz szükséges PhpMyAdmin adminisztrációs felületet. A XAMPP és a benne lévő programok lehetővé tették számunkra a lokális azaz a saját számítógépünkön való fejlesztést és tesztelést anélkül hogy szükségünk lett volna egy valós webkiszolgálóra. A kódbazis lefejlesztésére a Visual Studio Code nevezetű kód szerkesztőt használtuk. Ezen választás mögött a kód szerkesztő széles, a fejlesztést elősegítő és meggyorsító eszközök által. Eszközök mint például automatikus kód kiegészítés/ajánlás, naprakész frissítések, beépített git elérhetőség, fejlesztést megkönnyítő felhasználói felület, és még egyéb kiváló eszközök.

Felhasznált nyelvek

A program lefejlesztéséhez különféle nyelveket használtunk. A kódbazis legalapvetőbb nyelve a PHP. A hagyományos HTML-el ellentétben a PHP lehetővé tette számunkra a szerver oldali programozást mely rettentő fontos szempont hiszen az oldal felhasználói adatokat kezel. A PHP segítségével a HTML elemek fejlesztése is leszűkült hiszen a PHP dinamikusan generál HTML kódot melyet a kliens oldal azaz a felhasználó kap meg, azonban minden fontos háttérfolyamat a szerveren történik. A PHP mellett természetesen a HTML és CSS is nagy szerepet kap ott ahol fontos háttérfolyamat nem keletkezik. Ezen nyelveken kívül számos elem tartalmaz JavaScript logikát. A JavaScript a programunk esetében olyan, dinamikus vagy látványelemhez köthető folyamatokat intéz mely folyamatok lefuthatnak a kliens oldalon is hiszen különösen védett folyamatot nem végeznek.

Kiegészítő elemek

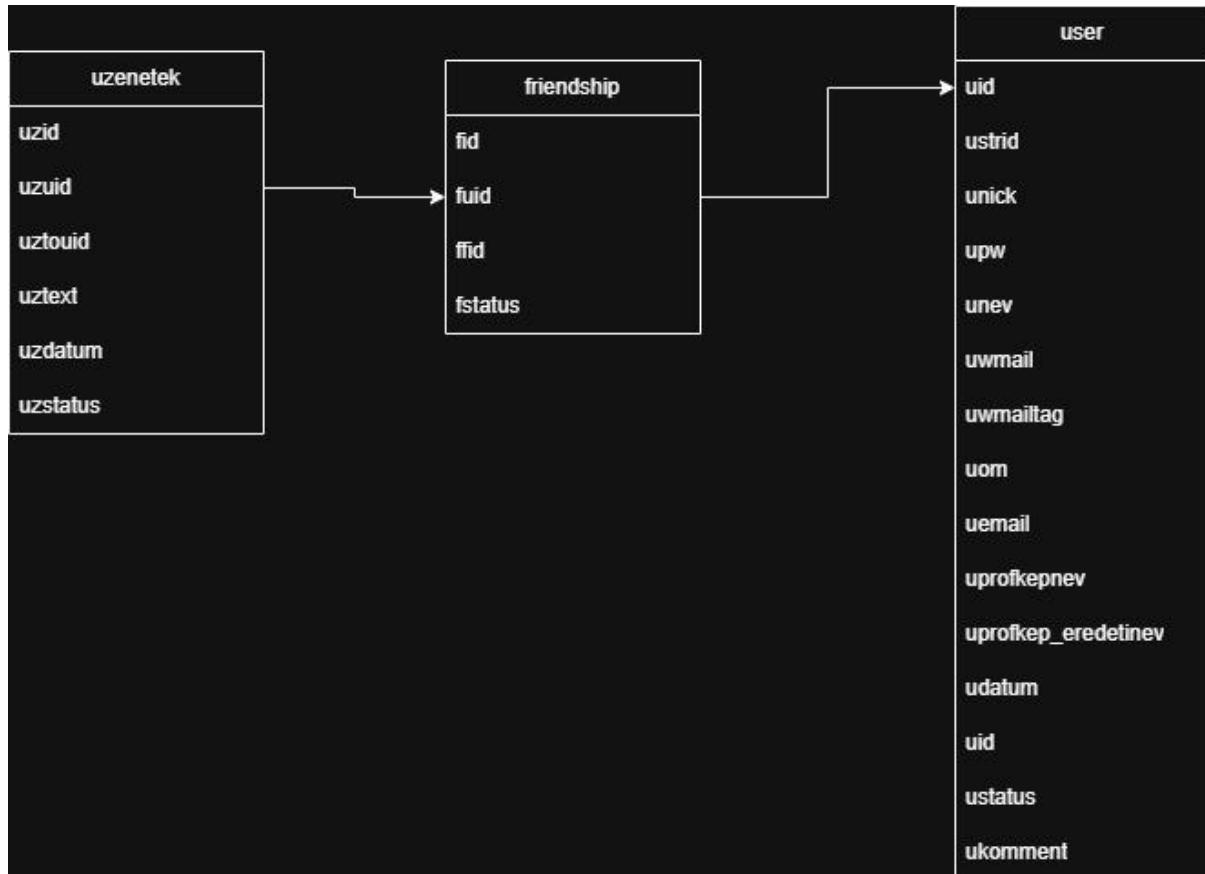
Megemlítésre méltó az is hogy a XAMPP által nyújtott programokat miért és miféleképpen használtuk fel. Az Apache szerver lényegében egy valós webszerver emulálásában segített, különösebb kifejtés nem szükséges. A MySQL azonban már érdekesebb, a MySQL mely egy adatbáziskezelő rendszer a weboldalon tárolandó adatok hatékony tárolására használtuk, választásának oka az is hogy egy, az évek során sokat tesztelt adatbázisról beszélünk így megbízható is. A PhpMyAdmin az adatbázis adminisztrációjára használtuk, ezen program a PHP támogatásának köszönheti a felőlünk érkező kiválasztást, ezen programmal hatékonyan kezelhattük az adatbázisba kerülő adatokat illetve az ott lévő adatok felhasználására is tökéletes lehetőséget és környezetet adott.

A fejlesztéshez használt hardver

A fejlesztés során meghatározó többségben az iskolában elérhető, iskolai számítógépeket használtuk, megállapítható tehát hogy a fejlesztés nem igényelt különleges erőforrásokat. Bizonyos esetekben az otthon elérhető számítógépeinket is használtuk azonban a helyzet változatlan.

Adatszerkezet

A weboldal adatbázisa számtalan táblát tartalmaz azonban a csevegő felület nem használja minden, a felhasznált táblákat és azok összekapcsolását az alábbi képen vehetjük szemügyre:



“friendship” tábla

friendship
fid
fuid
ffid
fstatus

A friendship (barátság) tábla követi nyomon hogy melyik felhasználó barátja melyik felhasználóknak. Amikor egy felhasználó hozzáad egy másik felhasználót a csevegő listájához akkor két új sor kerül a friendship táblába.

Fid:

Az fid a friendship tábla elsődleges kulcsa. Ebből fakadóan az elsődleges kulcs automatikusan generálódik.

Fuid:

Az fuid tárolja azt az id-t mely id (azonosító) a küldő félhez tartozik, azonban a az adatszerkezet kialakítása végett egy küldés során a fordítottja is bekerül egy új sorba. Nem mellesleg az fuid idegenkulcsként is szolgál

Ffid:

Az ffid azt az id-t tartalmazza mely id a fogadó félhez tartozik, azonban a az adatszerkezet kialakítása végett egy küldés során a fordítottja is bekerül egy új sorba.

Fstatus:

Az fstatus a két fél státuszát jelöli azaz P (pending, várakozó), ez a státusz addig marad fenn míg a másik fél (fogadó) nem küldött üzenetet vissza. Ha a fogadó fél is küldött vissza üzenetet a státusz A-ra (accepted, elfogadott) státuszra változik. A felhasználók letilthatják egymást, ebben az esetben a státusz B-re (blocked, blokkolt) státuszra változik ekkor a két fél nem képes egymásnak üzenetet küldeni. Fontos hogy az alapján hogy ki tiltott le kit, az fstatus különböző lesz, a letiltó fél ugyanis nem B hanem BB (blocked by, ...által letiltott) státuszt fog kapni. Ezzel biztosítva van hogy csak a letiltó fél legyen képes a tiltást feloldani.

“uzenetek” tábla

uzenetek	
uzid	
uzuid	
uztouid	
uztext	
uzdatum	
uzstatus	
uzfile	

Az “uzenetek” tábla minden, az üzenetekhez kapcsolódó adatot tartalmaz.

uzid:

Az ujid az üzenetek tábla elsődleges kulcsa. Ebből fakadóan az elsődleges kulcs automatikusan generálódik.

uzuid:

Az uzuid tárolja azt az id-t mely id (azonosító) az üzenetet küldő félhez tartozik. Idegenkulcsként is szolgál.

uztouid:

Az uztouid azt az id-t tartalmazza mely id az üzenetet fogadó félhez tartozik.

uztext:

Az uztext az elküldött üzenet szöveges tartalmát tárolja.

uzdatum:

Az uzdatum az üzenet küldésének pontos időtartamát tárolja (év-hónap-nap-óra-perc-másodperc).

uzstatus:

Az uzstatus az üzenet státuszát jelzi mely lehet S (sent, elküldött) vagy D (deleted, törölt).

uztext:

Az uzfile az elküldött fájlt tárolja.

Algoritmusok

Csevegések megjelenítése:

Ezen algoritmus a megfelelő csevegések megjelenítéséért felelős. Ez egy dinamikusan bővülő lista mely lekéri az adatbázisból azokat a felhasználókat akik között már létezik csevegés és megjeleníti azt a megfelelő felhasználóknak.

```
<div id="chatsFormContainer">
<form action="social_ir.php" method="POST" target="kisablak" class="chatsForm">

<?php
$query = mysqli_query($adb, "SELECT * FROM user INNER JOIN friendship ON user.uid = friendship.fuid WHERE ffid = '$_SESSION[uid]' AND fuid != '$_SESSION[uid]'");

while ($users = mysqli_fetch_assoc($query)) {
    $punick = $users['unick'];
    $puid = $users['uid'];

    echo '

        <input type="hidden" name="ffid" value="'.$puid.'"
        <input type="hidden" name="ffnick" value="'.$punick.'"

            <a class="ivlink" id="person" value="'.$punick.'" href="con.php?user='.$puid.'" target="chatTab">
                <h1 id="person_h1">'.$punick.'
            </a>
    ';

}
?>
</form>
```

A dinamikusan legenerált gombokra kattintva a megfelelő csevegést nyithatjuk meg és innentől már lehet üzenetet küldeni a következő képeken a dinamikus felület generálást és backend logikát láthatjuk:

```
<?php
echo '

<div class="chatFormContainer">

    <iframe name="chatTab" id="chatTab">

        </iframe>

    <script>
document.getElementById("chatTab").onload = function() {
    setTimeout(function() {
        var iframe = document.getElementById("chatTab");
        iframe.contentWindow.scrollTo(0, iframe.contentDocument.body.scrollHeight);
    });
};

</script>

    <script>
setInterval(function() {
    document.getElementById("chatTab").contentWindow.location.reload();
}, 900);
</script>

<br>
<div class="regContainer" id="chatSendContainer">

    <form onsubmit="clearInput()" action="social_ir.php" method="post" target="kisablak">

        <input type="text" placeholder="" id="message" name="message" autocomplete="off">

        <br><br>

        <button type="submit" id="chatformgomb">Küldés</button>

        <!-- Login link -->

    </form>

</div>
';

?>

<?php
session_start();

include("kapcsolat.php") ;

$query = mysqli_query($adb, "SELECT * from user WHERE uid != '$_SESSION[uid]' AND EXISTS (
    SELECT * FROM friendship WHERE ffid = uid
)");

while ($users = mysqli_fetch_assoc($query)) {
    $punick = $users['unick'];
    $puid = $users['uid'];
}

$message = trim($_POST['message']);

mysqli_query( $adb , "
    INSERT INTO uzenetek (uzid, uzuid , uztoid , uztext , uzdatum, uzfile)
    VALUES      (NULL, '$_SESSION[uid]' , '$puid','$message', NOW(), 'teszt.jpg')
    " ) ;

?>
```

Barát hozzáadása

A barát hozzáadása gombra nyomva egy másik panelre irányít át a program ahol az összes az adatbázisban szereplő felhasználót láthatjuk. Nevükre rákattintva hozzáadhatjuk őket mint barát és onnantól kezdve képesek vagyunk üzenetet küldeni illetve ők is képesek ugyan erre. Amikor egy falhasználó nevére rákattintunk amellett hogy beszélgetésbe kezdhetünk velük az ő nevük már nem lesz benne a barát hozzáadása listában hiszen az már megtörtént. Az alábbi képen látható a dinamikusan generált gombok melyek a felhasználókat tartalmazzák:

```
<div class="userSearchContainer">

<div>
<form action='socialsearch_ir.php' method='POST' target='kisablak' id="peopleForm">

<?php
$query = mysqli_query($adb, "SELECT * from user WHERE uid != '$_SESSION[uid]' AND NOT EXISTS (
|   SELECT * FROM friendship WHERE ffid = uid AND fuid = '$_SESSION[uid]'

)");
;

while ($users = mysqli_fetch_assoc($query)) {
    $punick = $users['unick'];
    $puid = $users['uid'];

    echo '
<form action="socialsearch_ir.php" method="POST" target="kisablak">
<input type="hidden" name="ffid" value="'.$puid.'">
|   <a href="#">
|       <button type="submit" class="ivlink" id="person" onclick=" setTimeout(function() {window.parent.location.reload()}, 20)">
|           <h1 id="person_h1">'.$punick.'
|       </button>
|   </a>
</form>
';
}

?>
</form>

</div>
```

Az alábbi képen pedig a kattintást követő esemény látható:

```
mysqli_query( $adb , "
    INSERT INTO friendship (fid , fuid , ffid , fstatus)
    VALUES
        (NULL, '$_SESSION[uid]' , '$_POST[ffid]', 'P')

    " ) ;

mysqli_query( $adb , "
    INSERT INTO friendship (fid , fuid , ffid , fstatus)
    VALUES
        ('$_POST[ffid]' , '$_SESSION[uid]', 'P')

    " ) ;

?>
```

Üzenet küldése

Miután a barát hozzáadása megtörtént mindenki fél képes egymásnak üzenetet küldeni. Amikor a felhasználó rákattint egy már a barátlistában lévő felhasználó nevére a csevegő felület megnyílik, legenerálódik. Ezután a felhasználó az ott található szöveges mezőbe írhatja üzenetét majd a küldés gombra vagy az enter billentyű (számítógépen) lenyomására elküldheti üzenetét. Ezen backend logikát láthatjuk a következő képen:

```
<?php  
echo '  
  
<div class="chatFormContainer">  
    <iframe name="chatTab" id="chatTab">  
    </iframe>  
  
    <script>  
        document.getElementById("chatTab").onload = function() {  
            setTimeout(function() {  
                var iframe = document.getElementById("chatTab");  
                iframe.contentWindow.scrollTo(0, iframe.contentDocument.body.scrollHeight);  
            });  
        };  
    </script>  
  
    <script>  
        setInterval(function() {  
            document.getElementById("chatTab").contentWindow.location.reload();  
        }, 900);  
    </script>  
  
<br>  
<div class="regContainer" id="chatSendContainer">  
    <form onsubmit="clearInput()" action="social_ir.php" method="post" target="kisablak">  
        <input type="text" placeholder="" id="message" name="message" autocomplete="off">  
        <br><br>  
        <button type="submit" id="chatformgomb">Küldés</button>  
        <!-- Login link -->  
    </form>  
  
</div>  
';  
?  
>
```

Fejlődési lehetőség

Merre tovább?

A csevegő felület tömérdek fejlődésen mehet keresztül. Amellett, hogy egy program normális esetben sosem készül el teljesen vannak olyan pontok, ahol ezen csevegő program erőteljesen képes a fejlődésre, illetve mely területeken érdemes mindenképpen fejlődést előidézni.

Működési alapelvek

A program jelenleg JavaScript ablak frissítéssel tölti be a beérkező üzeneteket mely így folyamatos időnkét enyhén irritáló villogást is okozhat, habár ez böngészőtől is függhet, illetve az idő nyomós részében nem különösebben túl irritáló. Azonban annak érdekében, hogy a UX (User Experience / felhasználói élmény) kényelmesebb és hatékonyabb legyen, az olyan technológiák mint az AJAX vagy WebSocket implementálása elengedhetetlen.

Csoportok kialakítása

A fejlesztés során a csoportok kialakítása bizonyos tényezők miatt (melyek között külsős, a fejlesztők hatáskörén kívül eső problémák is felmerültek) a csoportok kialakítása nem került lefejlesztésre. A számunkra elérhető adatok helyenként hiányosnak bizonyultak ebből fakadóan a csoportok implementálása egy jó fejlesztési lehetőség amint egy megfelelőbb adatrendszer ki tud alakulni mely számunkra még nem volt elérhető.

Teljes modernizálás

Az oldal PHP-ra épült mely bár egy jól ismert és stabil nyelv az évek során már-már elavultnak számít. Manapság már a Node.js segítségével például lehetőségünk nyílik szerver oldalon programozni JavaScriptben mely így egy gyorsabb élményt biztosít. Elsősorban, a továbbfejlesztés elérhető azzal is, hogy a PHP nyelvnél maradva az oldalt továbbfejlesztjük valamiféle PHP keretrendszerrel (pl Laravel) ebben az esetben egy összeszedetebb és erősebb programot kapunk. Azonban, ha ténylegesen modernizálni szeretnénk JavaScript alapú keretrendszerek hatékonyabb fejlesztési lehetőségeinek számítanak. Ezen esetben akár használhatunk React-ot, Vue-t vagy egyéb keretrendszert de akár a React-native segítségével még egy szintet léphetünk és egyenesen fejleszthetünk webre és mobiltelefonra azonban így már a mobiltelefon verzió már applikációként fog futni így még hatékonyabb mobil élményt adhatunk felhasználóinknak.

Tesztelői hozzáférés

A teszteléshez szükségszerűen egy megfelelő engedélyekkel rendelkező felhasználót hoztunk létre, ennek adatait itt találhatjuk, ezen felhasználó segítségével a funkciók mindegyike kipróbálásra kerülhet.

1. Felhasználónév: teszt
Jelszó: teszt123
2. Felhasználónév: tesztdok
Jelszó: teszt123

Felhasználói dokumentáció

A weboldal célja

Kedves Felhasználó! A jelen dokumentáció egy közösségi célú webes platform használatát ismerteti, amelyet elsősorban egy **iskolai közösség** számára fejlesztettünk. A weboldal célja, hogy elősegítse a diákok közötti kommunikációt, tájékozódást, véleménynyilvánítást és a közösségi élet szervezését. A rendszer különbséget tesz az **általános felhasználók** és a **diákönkormányzati tagok** között. Bizonyos funkciók – például a szavazás indítása vagy a hivatalos posztok közzététele – kizárolag a DÖK tagok számára elérhetők. A jogosultságokat a rendszer automatikusan ellenőrzi. Ezzel ellentétben a csevegési felületen nincs megkülönböztetés, mindenki egyenjogú.

A legfontosabb funkciók a következők:

- Bejegyzések olvasása és hozzászólás írása:** A felhasználók hozzászólhatnak mások tartalmaihoz, azazzal megbízott személyek pedig új bejegyzéseket is írhatnak.
- Fájlmegosztás:** A rendszer lehetőséget biztosít fájlok feltöltésére, amelyeket más felhasználók letölthetnek.
- Szavazások indítása és részvétel:** A diákönkormányzati (DÖK) tagok a bejegyzésekben szavazásokat hozhatnak létre, amelyekben bárki részt vehet.
- Chat funkció:** Az oldalon bárkivel lehetséges élő csevegni,
- Csoportos beszélgetés:** Osztálycsoport létrehozható osztálytársak között feltéve, ha a regisztrált felhasználók ugyan azt az osztályt adják meg a profil oldalon.

Program használatához szükséges eszköz

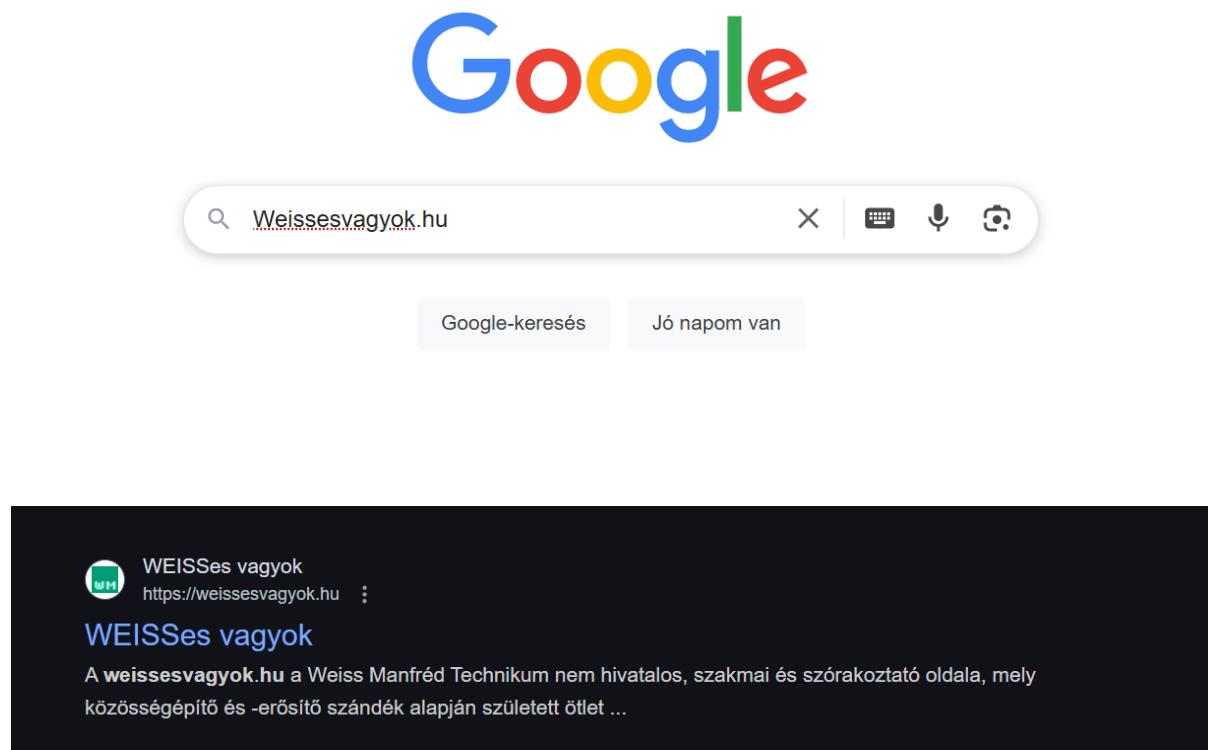
A platform számítógépről, laptopról, tabletről vagy okostelefonról is elérhető, és böngészőn keresztül működik, tehát **nem szükséges külön telepíteni**, elegendő internetkapcsolattal rendelkezni.

Program elindítása

A program elindításához szükség van egy böngészőre, majd a böngészőn belül kétféleképpen érhetjük el a programot.

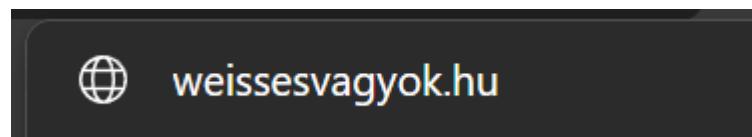
Egyszerű keresés:

Egyszerű keresés a böngészőben való keresést jelenti, amit (remélhetőleg) a találatok között meg is találunk.



Keresés URL alapján:

URL alapján történik a keresés, ha a weboldal (általában) neve elő „http://” vagy „https://” protokoll kerül, majd jelen esetben „.hu” a végére. A legfrissebb keresők, melyek a böngésző tetején levő sávban helyezkednek el már automatikusan kirakják a megfelelő protokollt, ezzel is megkönnyítve a pontos keresést.



Az oldalra érkezés után:



MÉRÉSEK TESZTEK PORTRÉK PÁLYAÍV



Kezdőlap

A weissevagyok.hu a Weiss Manfred Technikum nem hivatalos, szakmai és szórakoztató oldala, mely közösségegítő és eredményes diákokat szolgáló színtér.

Az oldal célja a formálás, a kialakulás fázisában van, az új tanulmák felülvizsgájának előkészítésére.

A tervben tartalmuk között szerepelnek:

- közösségi keretek, például tanételek az iskolához, az oktatás aktualitásairól;
- érdemelhető önműenneti tesztek, melyek részére mutatunk akár a pályá- és szakmai vállalkozások körülmenetben is;
- portrék, illetve interjúk, beszégtörökök az iskolai tanirányával, segítő pedagógusainak;
- korábbi tanulmányi eredmények leírásai bemutatói iskolák, tanulmányi eredmények, munkába általában szóbeli tapasztalatuk;
- és természetesen minden levélről, amit kapunk, vagy küldünk.

[Látssuk Google Maps-en!](#)



Kezdőlap

A weissevagyok.hu a Weiss Manfred Technikum nem hivatalos, szakmai és szórakoztató oldala, mely közösségesítő és eredményes diákokat szolgáló színtér.

Az oldal célja a formálás, a kialakulás fázisában van, az új tanulmák felülvizsgájának előkészítésére.

A tervben tartalmuk között szerepelnek:

- közösségi keretek, például tanételek az iskolához, az oktatás aktualitásairól;
- érdemelhető önműenneti tesztek, melyek részére mutatunk akár a pályá- és szakmai vállalkozások körülmenetben is;
- portrék, illetve interjúk, beszégtörökök az iskolai tanirányával, segítő pedagógusainak;
- korábbi tanulmányi eredmények leírásai bemutatói iskolák, tanulmányi eredmények, munkába általában szóbeli tapasztalatuk;
- és természetesen minden levélről, amit kapunk, vagy küldünk.

[Látssuk Google Maps-en!](#)

Telefonos megjelenés



Az oldal megnyitásakor a kezdőlap üdvözöl minket, majd léphetünk tovább a felső opciók egyikével. Ez telefonos nézetben megváltozik, és ilyenkor a jobb oldali menüsáv nem jelenik meg automatikusan, hanem egy kis három vonalas ikon (más néven „hamburger menü”) jelenik meg az oldal jobb felső részén.



A bejelentkezés gombra kattintva beléphetünk már meglevő fiókunkba, ha pedig nincs, akkor ugyan erről a pontról tovább lépve regisztrálhatunk egy újat. A regisztrációhoz mindössze egy felhasználónévre, egy email címre, továbbá egy jelszó, illetve jelszó megerősítésére van szükség.

Bejelentkezve a rendszer a nevünkön üdvözöl minket, majd lehetőség biztosít a többi funkció eléréséhez.

Bejelentkezés után:



MÉRÉSEK

TESZTEK

PORTRÉK

PÁLYAÍV

Bejelentkezés után a menüsor kiegészül egy lenyíló elemmel melyet, ha megnyitunk különböző kiegészítő tartalmakat láthatunk, amik csak bejelentkezett felhasználóknak elérhető. Ugyan ez igaz a profil oldalra is, amely jobb fent érhető el, a megadott névre kattintva. A név mellett a kijelentkezést láthatjuk.



WEISSES vagyok



Pszichológus

Adok-Veszek

Social

DÖK

dad

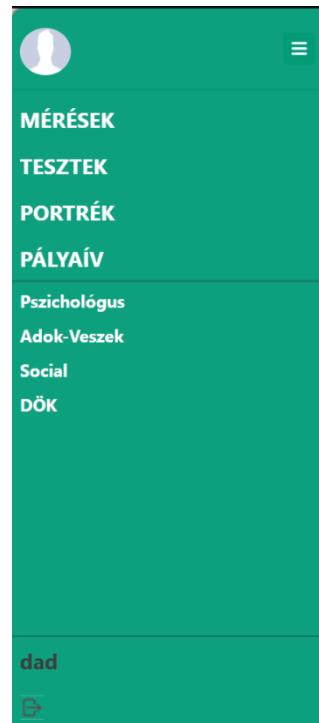


Amikor az oldalt mobiltelefonról vagy kisebb képernyőmérőtű eszkösről nyitjuk meg, az automatikusan úgynevezett telefonos (vagy mobilbarát) nézetre vált. Ez azt jelenti, hogy a felület elrendezése alkalmazkodik a kisebb képernyőhöz, hogy minden funkció kényelmesen elérhető maradjon. Fontos tudni, hogy a menü viselkedése teljesen megegyezik az asztali nézetével, csak a megjelenése más. minden funkció ugyanúgy működik, a különbség csupán a vizuális elrendezésben rejlik.

Egyes esetekben, például ha egy laptop vagy asztali számítógép képernyője kicsi, vagy ha a böngésző ablakot erősen lekicsinyítettük, a rendszer szintén a mobilos nézetet jelenítheti meg. Ez megtévesztő lehet, hiszen ilyenkor nem mobilról, hanem számítógépről használjuk az oldalt.

Ha azt tapasztaljuk, hogy kevesebb funkció látszik, vagy például a jobb oldali menü nem jelenik meg automatikusan, akkor a következőket tehetjük:

- **Próbáljuk meg az oldalt „kizoomolni”:** Ez azt jelenti, hogy kisebb méretben nézzük az oldalt. Ezt általában a Ctrl gomb lenyomva tartása mellett az egér görögjének lefelé tekerésével tehetjük meg, vagy a Ctrl + - billentyűkombinációval.
- **Növeljük meg a böngésző ablak méretét:** Húzzuk ki az ablak sarkát, hogy nagyobb legyen a látható felület.



Felhasználói dokumentáció: Közösségi média

Az oldal elérése

A korábban említett menüből a DÖK opció kiválasztásával, amennyiben a felhasználó korábban a profil oldalon megadta az osztályát, automatikusan átirányítást kap a felületre, ellentétben egy figyelmeztetést kap, illetve egy tovább irányítást az adatlapra. Ez a felület minden felhasználó számára elérhető, azonban bizonyos funkciók kizárolag a diákönkormányzati tagok számára használhatók.

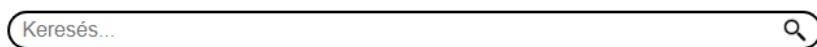
Általános jogosultságok

Az átlagos felhasználók ezen az oldalon nem csupán olvasási jogosultsággal rendelkeznek, mert amellett, hogy megtékinthetik a közzétett tartalmakat, akár részt vehetnek szavazásokban és hozzászólhatnak a bejegyzésekhez is.

Kereső funkció

A weboldalon elérhető kereső lehetőséget nyújt a bejegyzések közötti gyors szűrésre. A funkció célja, hogy a felhasználók egyszerűen megtalálják a számukra releváns információkat, különösen akkor, ha már sok tartalom gyűlt össze. A keresőmező az oldal tetején, jól látható helyen található.

A keresés a bejegyzések közt történik. A keresés nem érzékeny a kis- és nagybetűkre, valamint részleges egyezés alapján is találatot ad, így már néhány karakter beírása után is megjelenhetnek kapcsolódó bejegyzések.



Találatok a következőre: Kettő

 dad 2025-05-17 11:21

egy kettő három



Hozzászólás írása

Hozzászólást az egyes bejegyzések alatt lehet írni, a beviteli mezőben. A hozzászólás megírása után a mező jobb oldalán található gomb megnyomásával, vagy az Enter billentyű lenyomásával lehet azt elküldeni.

 dad 2025-03-10 02:28

aaaaaaaaaaaaaa



 dad 2025-04-16 02:43

b



A mezőre kattintáskor a háttér fehérre változik, majd megjelenik az írást segítő kurzor.



Sikerességek után az oldal lefrissül, majd a hozzászólás azonnal megjelenik a bejegyzés alatt, időbényeggel és a hozzászóló nevél ellátva.

Szavazás lehetősége

A szavazás leadásához a bejegyzés jobb oldalán található opciókra kell kattintani, amelyek egy-egy válaszlehetőséget jelentenek. Ezek az opciók rádiógomb formájában jelennek meg, tehát egyszerre csak egy válasz választható ki. A felhasználónak ki kell jelölnie azt az opciót, amelyre szavazni szeretne, majd az oldal automatikus frissítés következtében meg is jeleníti a friss adatokat.

The screenshot shows a mobile application interface for voting. At the top left is a user icon labeled "dad" and the date "2025-05-11 08:14". Below this is a list item with the letter "b". To its right are two buttons for voting: one for option "a" (which is selected) and one for option "b". Each button has a red border and displays the number of voters: "1 szavazó" for "a" and "0 szavazó" for "b". At the bottom right is a large, light blue rectangular button with a right-pointing arrow.

A rendszer lehetővé teszi az újraszavazást is, így ha valaki meggondolja magát, később módosíthatja a döntését. A korábban kiválasztott opció helyett új opciót jelölhet be, és ismét rögzítheti azt.

Telefonos nézet

A telefonos nézet esetében az opciók egymás alá csúsznak, de a működési elve ugyan úgy működik, mint a weboldal nagy képernyős változatán.

This screenshot shows a mobile application interface for voting, similar to the previous one but with a different layout. It features a user icon "dad" and the date "2025-05-11 08:14". Below is a list item with "b". To the right are two buttons for "a" and "b", each with a red border. Both buttons show the voter count: "1 szavazó" for "a" and "0 szavazó" for "b". A large light blue arrow button is at the bottom right.

Saját hozzájárulások törlése

A törlés opció akkor válik elérhetővé, ha a hozzájárulás jobb felső sarkában található három függőleges pont („:”) ikonra kattintunk. Ez a gomb egy lenyíló menüt jelenít meg, amelyben jelenleg csak a Törlés gomb megtalálható.

Fontos megjegyezni, hogy az ikon csak akkor jelenik meg, ha a hozzájárulás a saját nevünkhez tartozik, tehát mi írtuk azt. Ez biztosítja, hogy mások hozzájárulásait ne tudjuk eltávolítani, csak a sajátunkat.

A „Törlés” opcióra kattintva egy megerősítő kérdést kapunk, amelyben a rendszer rákérdez, valóban szeretnénk-e törölni a hozzájárulást. Ezzel elkerülhető a véletlen adatvesztés. Ha megerősítjük a törlést, a hozzájárulás eltávolításra kerül az oldalról, és a megjelenített bejegyzés alatt már nem fog látszódni.

This screenshot shows a mobile application interface with a delete confirmation dialog. At the top right is a green button labeled "Törlés". In the center is the text "Ez egy teszt bejegyzés?". The background shows a user icon "dad" and the date "2025-04-14 04:10".

Letöltés

A fájlok letöltése szintén bárki számára elérhető, amennyiben az adott bejegyzéshez tartozik fájl. Ezek a fájlok a bejegyzés alsó részén jelennek meg, közvetlenül a szavazási lehetőségek alatt – ha szavazás is tartozik a bejegyzéshez. Amennyiben nincs szavazás, a fájl(ok) még közelebb helyezkednek el a bejegyzés szövegéhez.

A fájlok egy-egy kattintható gombként jelennek meg, rendszerint az eredeti nevükkel feliratozva, így könnyen azonosítható, hogy melyik fájl mit tartalmaz. A letöltéshez elegendő rákattintani az adott fájl nevére vagy az azt jelző gombra. Ekkor a böngésző – típusától és beállításaitól függően – automatikusan elindítja a letöltést, vagy lehetőséget kínál a fájl megnyitására, mentésére.

Fontos megemlíteni, hogy a letöltés nem változtatja meg az oldal URL-címét, és nem nyit meg új oldalt sem, így a felhasználó minden a bejegyzésnél marad. Ez különösen kényelmes mobil eszközökön és kisebb kijelzőkön, ahol az egyszerű, zökkenőmentes navigáció kiemelt jelentőséggel bír.

dad 2025-04-10 15:58

Teszt

Opció 1	0 szavazó
Opció 2	2 szavazó
Opció 3	0 szavazó

[↓ 1662476584_james-ghio-asset.jpg](#) [↓ download.jpg](#) [↓ 1647575286_fnel.png](#)

[↓ 1675727298_IMG_.png](#)

Telefonos nézet

Telefonon a fájlok szintén az opciók alá kerülnek, bár 3 helyett csak 2 opció jelenik meg soronként.

dad 2025-04-10 15:58

Teszt

Opció 1	0 szavazó
Opció 2	2 szavazó
Opció 3	0 szavazó

[↓ 1662476584_james-ghio-asse...](#) [↓ download.jpg](#)

[↓ 1647575286_fnel.png](#) [↓ 1675727298_IMG_.png](#)

Diákönkormányzatos jogosultságok

Az oldal speciális felhasználói jogkört biztosít az arra jogosult diákok számára. Ez a jogosultság egyfajta kiterjesztett hozzáférést jelent a rendszerben, mely több lehetőséget biztosít, mint az átlagos felhasználói fiókok esetén.

Fontos, hogy a diákoknak jogosultság nem automatikusan jár a felhasználóknak. Ezt külön szükséges adminisztratív úton beállítani – például egy rendszergazda vagy iskolai koordinátor jóváhagyásával, aki DÖK taggá minősíti az adott felhasználót. A felület ezt követően azonnal biztosítja az említett extra lehetőségeket.

Új bejegyzések létrehozása

A felületen lehetőség van nyilvános bejegyzések kiírására, melyeket az összes felhasználó láthat. Ezekhez a bejegyzésekhez csatolhatnak szöveget, fájlokat, valamint szavazási lehetőségeket is, továbbá a bejegyzést közelgő eseményként.

A bejegyzést kitenni a **közzététel** gombbal lehetséges, amennyiben minden annak megfelelően lett kitöltve.

Szöveges beviteli mező

A mezőre kattintva a mérete nagyobb lesz, ezzel is biztosítva a jobb átláthatóságot és szerkeszthetőséget.

A szöveg úgy kerül megjelenésre, ahogy az le lett írva. Magyarán sortörés esetén sortöréssel jelenik meg, emojik használatával emojikkal. Kivételt képez az üres, vagy csak szóközt tartalmazó bejegyzés, illetve az 1-nél több szóköz használata. Ezeket a rendszer külön kezeli, az üres bejegyzést pedig kirakni sem engedi.

Szavazás létrehozása

A „**Szavazás létrehozása**” gombra kattintva egy felugró ablak jelenik meg. Ez az ablak tartalmazza azokat a mezőket, ahová a szavazási opciókat kell beírni. minden sor egy-egy válaszlehetőséget jelent. A rendszer minimum két megadott opciót igényel, ellenkező esetben a rendszer közzétételkor visszadobja. Az ablak dinamikusan kezeli a válaszlehetőségek beviteli mezőit, ezzel is támogatva a kényelmes és gyors használatot.

- Ha a felhasználó például a harmadik, utolsóként látható mezőre kattint, a rendszer automatikusan létrehoz egy újabb (negyedik) mezőt a meglévők alá. Ez a működés folytatható, vagyis minden alkalommal, amikor az éppen utolsó mező tartalommal bővül vagy fókuszt kap, új, üres mező kerül kihelyezésre a lehetőségek kiterjesztése érdekében.
 - Ezzel szemben, ha a felhasználó például a második mezőre kattint vissza, vagyis nem az utolsó mezőre, akkor nem történik új mező kihelyezése.
- A rendszer csak akkor ad új mezőt a listához, ha a sorban utolsó, látható mezőt használják, és az legalább részben ki van töltve.
- Fontos, hogy meglévő mezők nem törlődnek automatikusan visszalépéskor. Tehát ha például a második mező már tartalmaz szöveget, és a felhasználó visszatér az első mezőhöz, az nem okozza sem mezők eltűnését, sem újak létrejöttét.

A megadható opciók karakterhossza legfeljebb 26 lehet. Amennyiben a felhasználó hosszabb szöveget próbál bevinni, a további karaktereket nem engedi begépelni.

A mezők kitöltése során nem szükséges minden sort használni: az üresen hagyott sorokat a rendszer figyelmen kívül hagyja, és a közzététel során automatikusan törli. Ha a felhasználó például öt sorból csak hármat tölt ki, a másik kettő nem jelenik meg a szavazásban.

A rendszer ezen kívül önműködően javítja a fölösleges szóközöket is, például ha valaki véletlenül több szóközt ír egymás után, vagy a sor elejére vagy végére tesz felesleges üres karaktereket, a rendszer ezeket automatikusan eltávolítja.

A közzétételt követően a szavazás a bejegyzés részévé válik, és minden felhasználó számára megjelenik. Ők egyetlen kattintással leadhatják szavazatukat az általuk választott opcióra. A rendszer minden felhasználónak csak egyetlen szavazatot engedélyez, és azt követően az adott bejegyzésen belül a szavazás eredményeit is megtekinthetik.

Opciók megadása:

Opció 1
Opció 2
Opció 3
Opció 4
Opció 5

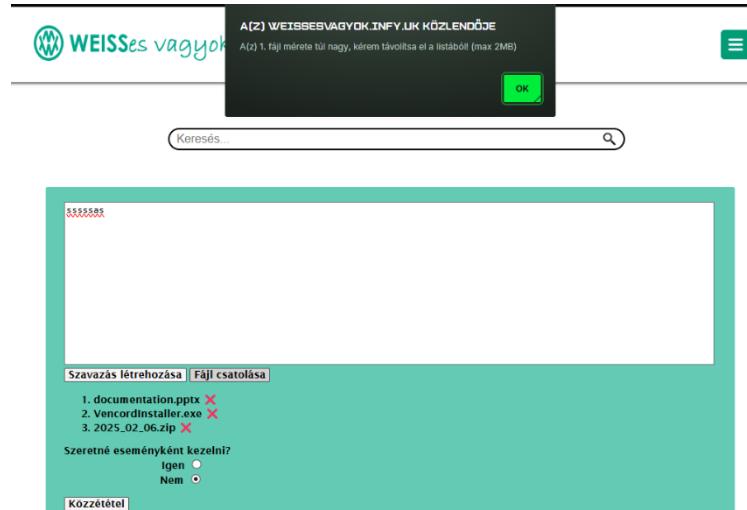
Megjegyzés:
Abban az esetben, ha nem szeretné szavazást létrehozni, vagy megsinced szüksége egyes mezőkre, hagyja őket üresen!

Kész

Fájl csatolása

A rendszer egyszerre több fájl csatolását is támogatja, azonban a csatolható fájlok mérete korlátozva van 2MB méretű állományokra. Ezt túllépve a program közzétételkor figyelmeztet és nem hagyja jóvá a bejegyzés posztolását.

Fájlméret túllépése esetén a rendszer felhívja figyelmünket a hibára, illetve annak sorszáma. Több problémás fájl esetén bár egyesével írja ki, de minden pontosan mutat rá a fájl számára, amelyet a piros X megnyomásával tudunk törölni a listából.



A kiválasztott fájl(ok) neve a feltöltés után megjelenik a bejegyzésmező alatt, vizuálisan is jelezve, hogy azok hozzá lettek csatolva. A csatolt fájlokat a rendszer a bejegyzéssel együtt menti el, és azok a közzététel után a bejegyzés alján (szavazás alatt), a többi felhasználó számára is láthatóvá és letölthetővé válnak.

Eseményként kezelés

A bejegyzések létrehozásakor lehetőségük van a tartalmakat **eseményként** megjelölni, amennyiben az adott bejegyzés valamelyen konkrét időponthoz, eseményhez vagy határidőhöz kapcsolódik (pl. kirándulás, szalagavató, határidős feladat, gyűlés). Az eseményként megjelölt bejegyzések külön megjelenítést kapnak a weboldalon:

- Kizárolag nagyobb képernyőn (minimum 1210 pixel szélesség felett) látható egy külön események oszlop vagy szekció az oldalon, amely listázza az összes aktív eseményt időrendi sorrendben (legfrissebb legfelül).
- A bejegyzés szövegében továbbra is megjelenik az esemény leírása, csatolt fájlok és szavazás lehetőségek, de az esemény státusz miatt gyorsabban felismerhető és visszakereshető.

Az eseményként való megjelölés egy egyszerű kérdés formájában jelenik meg: „**Eseményként szeretnéd megjelölni?**”. Az „**Igen**” opció kiválasztásakor a rendszer bekér egy dátumot. A dátum megadható manuálisan vagy a naptár ikon segítségével is (a felület viselkedése böngészőtől függően változhat). A program csak aznap vagy jövőbeli dátumot fogad el, a korábbi időpontokat nem engedi megadni.

A dátumot beállítva mást nem kér a rendszer, automatikusan törölni fogja az események sávból, amint a dátum változik, azaz aznap éjfélig lesz elérhető. Amint ez bekövetkezik, úgy már csak a bejegyzések között lesz megtekinthető.

Osztályok megemlítése

A beviteli mezőbe beírva az osztály évfolyamát és csoportját ponttal (.) elválasztva a program leellenőrzi valóban létezik-e ez az osztály. Ha nem, azt egy piros X jelzi, ha igen, egy zöld ✓.

Több osztály megadása esetén egy hagyományos vessző és egy szóköz használata szükséges.

Egyezés esetén a bejegyzés közzétételekor rendszerüzenetet is kapnak a felhasználók a beszélgető felületen a bejegyzés létrejöttéről.

Ha nincs egyezés, akkor értesíti a felhasználót a hibáról, és megvárja, míg azt javítja, vagy teljes egészében törli.

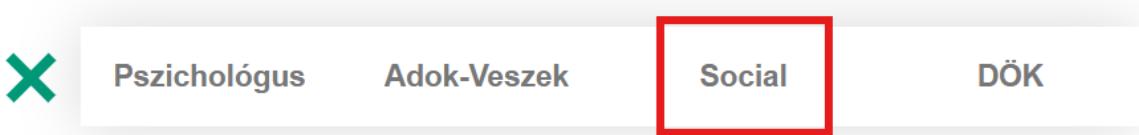
Bejegyzések és hozzájárások kezelése

A bejegyzések létrehozásán túl a felhasználóknak lehetőségük van saját bejegyzéseik törlésére, amennyiben azokat ők hozták létre. Ez a funkció a bejegyzés jobb felső részén található menü megnyitásával érhető el, ahol megjelenik a **Törlés** opció.

A hozzájárások esetében szintén rendelkezésre áll a törlés lehetősége. Csak úgy, mint az átlagfelhasználóknál, a saját hozzájárásait bármikor eltávolíthatják, és ezen felül mások hozzájárásainak törlésére is képesek, amennyiben az adott hozzájárás problémás, szabálytalan, vagy nem megfelelő, illetve a problémás személy nem rendelkezik diákokkormányzati jogosultsággal.

Felhasználói dokumentáció: Csevegő felület

A "social" menüpontot kiválasztva érkezhetünk meg a csevegőfelületre.

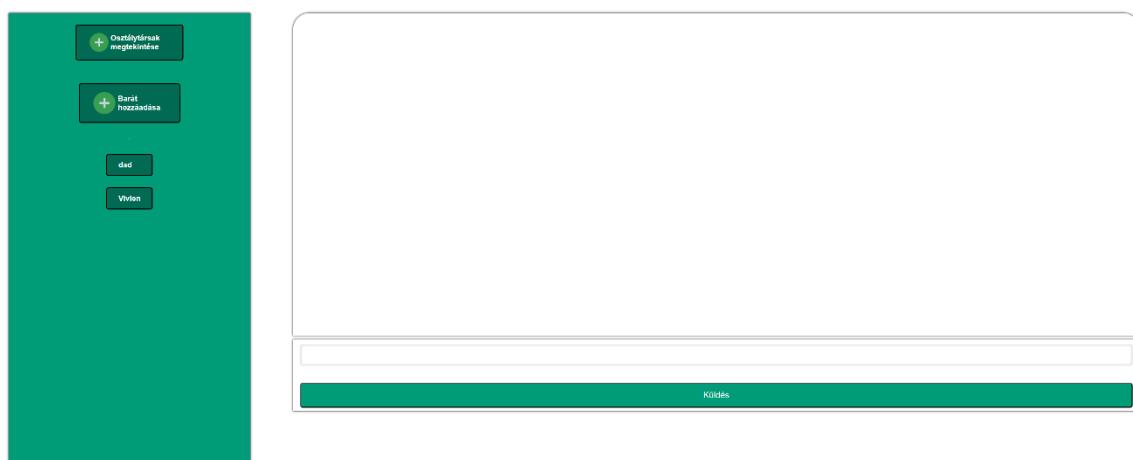


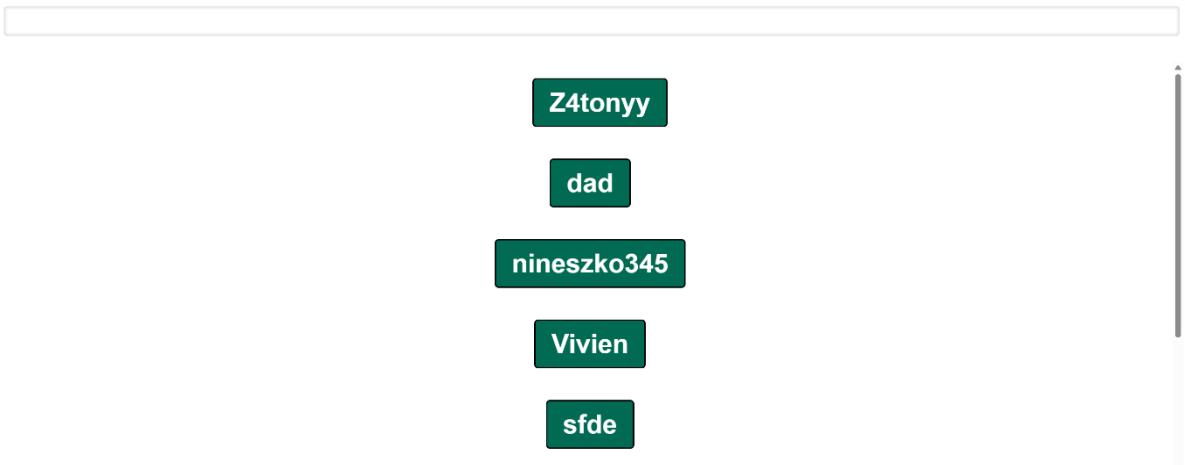
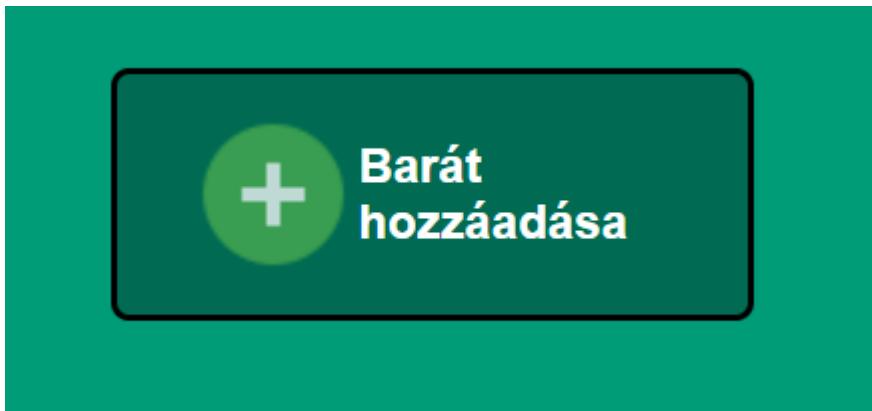
Mi is ez a program?

Ez a program egy csevegőfelület a Weissesvagyok.hu oldalán. Tehát egy olyan programról beszélünk melyen keresztül üzeneteket küldhetünk más felhasználóknak. Az egyértelmű üzenetküldés mellett még egy elengedhetetlen funkció a barát felvétele.

Program használata

Miután beléptünk és a csevegőfelületre navigáltunk alapértelmezetten (természetesen ha új felhasználók vagyunk) nincs egyetlen barátunk sem, ekkor a barát hozzáadása gombra kattintva tekinthetjük meg az oldalon elérhető felhasználókat.

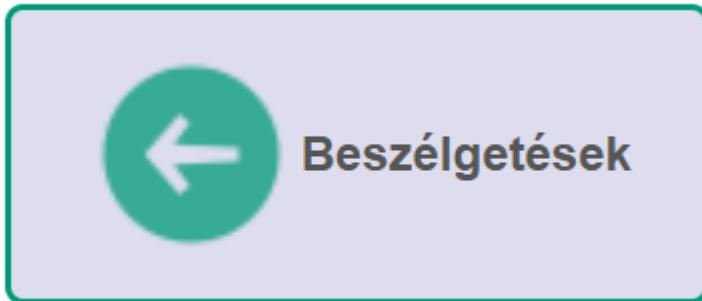




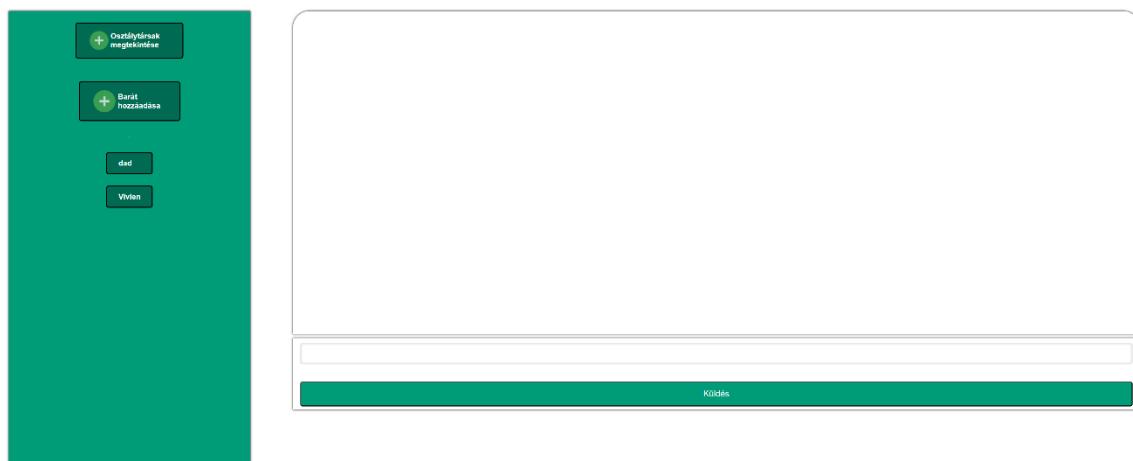
Ha üzenetet szeretnék küldeni akkor elsősorban hozzá kell adnunk a barátlistához azon felhasználókat akikkel beszélgetést szeretnénk folytatni. Egy barát hozzáadásához a barát hozzáadása gomb megnyomása után betöltődő listában megjelenő nevekre kell rányomnunk. Ha megtaláltuk az a felhasználót akit szeretnénk barátlistánkhoz adni a nevére kattintással adhatjuk őt hozzá barátlistánkhoz mely a beszélgetés elkezdéséhez elengedhetetlen. A barát hozzáadása felületen található egy kereső sáv is mely a kellő felhasználók megtalálását megkönnyíti és meggyorsítja, ha már tisztában vagyunk a keresett felhasználó felhasználónevével.

A barát hozzáadás után:

A beszélgetések gombra kattintva visszaléphetünk beszélgetéseinkhez. Mely szükséges hogy láthassuk barátlistánkat illetve elkezdhessük a velük való beszélgetést.

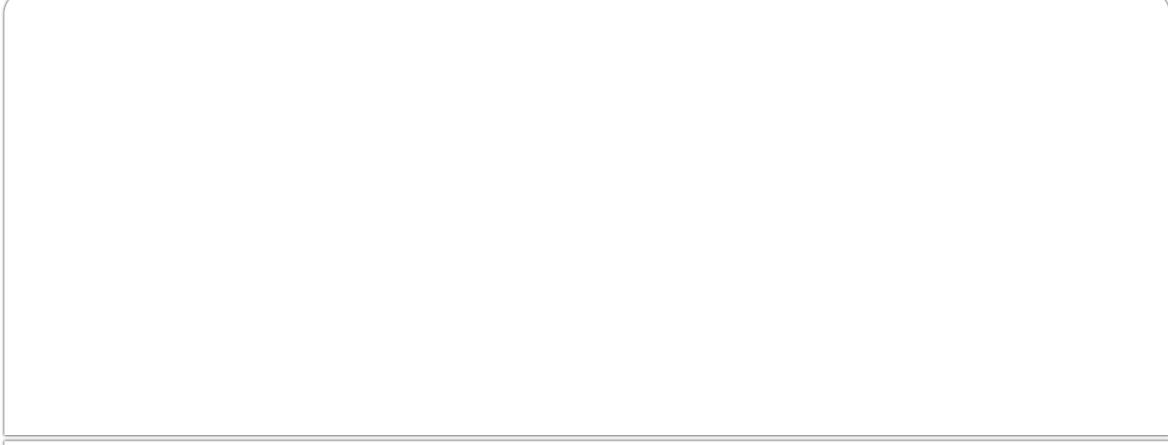
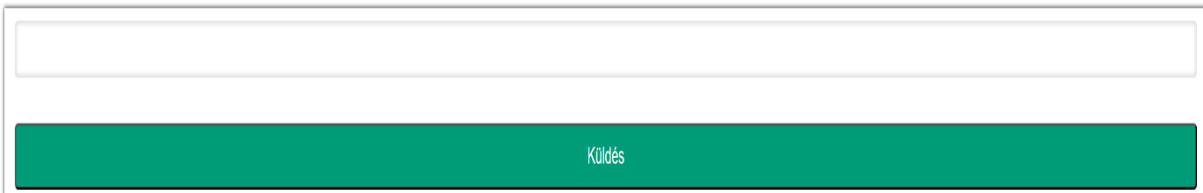


Ha visszaléptünk a beszélgetésekbe a barátlistánk a bal oldalon lelhető fel, itt azon felhasználók vannak, akiket vagy mi hozzáadtunk listánkhöz, vagy azok, akik minket hozzáadtak a saját listájukhoz. Most már a listában fellelhető felhasználókkal képesek vagyunk beszélgetést kezdeményezni vagy folytatni. A megfelelő felhasználó nevére kattintva megnyithatjuk a beszélgetést.



A jobb oldali fehér terület (mely első interakciónál üres, mint ahogyan a képen is látható!) az a terület mely az üzenetek megjelenítésére szolgál. Az alatta lévő üres terület pedig egy bemeneti mező mely az üzenetünknek van dedikálva. Ahhoz, hogy üzenetünket elküldhessük ebbe a mezőbe kell kattintani és beírni az elküldeni kívánt üzenetet. Ezen bemeneti mező rendelkezik egy limittel hiszen még ha csak szövegről is van szó az adatbázisban ez helyet foglal. A limit 2000 karakter!

Az üzenet beírása után természetesen szeretnénk elküldeni üzenetünket (optimális esetben) ezt az “enter” billentyű vagy a “küldés” gombra kattintva tehetjük meg.

A large, empty rectangular area with rounded corners, intended for the user to type their message. It occupies the top half of the main window.A form interface consisting of two horizontal sections. The top section is a white input field for the message. The bottom section is a solid teal-colored button with the word "Küldés" (Send) written in white capital letters.

Összefoglalás

Értékelés:

A projekt megvalósítása során lehetőségünk nyílt arra, hogy elméleti tudásunkat gyakorlati tapasztalattá alakítsuk. A fejlesztés során különösen nagy hangsúlyt fektettünk a felhasználói élményre, az egyszerű kezelhetőségre, a multiplatform elérhetőségre, valamint arra, hogy az oldal minden fontos funkcióval ellássa a célcsoportot.

A munka során számos kihívással szembesültünk, például az adatkezeléssel, a dinamikus megjelenéssel, vagy a jogosultságok pontos meghatározásával. A projekt megoldása jelentős mértékben hozzájárult szakmai fejlődésünkhez, különösen a PHP, JavaScript, HTML és CSS nyelvek terén. A funkciók működésének és hibakezelésének átgondolt kialakítása során fejlesztettük problémamegoldó képességeinket is.

Nagy segítség volt, hogy a közös célnak hála tarthattunk eszmecserét, kikérhettük egymás véleményét, kitalálhattuk az adatbázis rendszerek felépítését és érvelhettünk egymás elképzelései ellen, esetleges hibákat kiküszöbölvé.

Bár mindig van hová fejlődnünk, úgy érezzük, sikerült a legtöbb elképzelésünket megvalósítani, és egy felhasználóbarát rendszert létrehoznunk, amely valóban segítheti a diákok közötti kommunikációt és a diákkormányzati munka szervezését.

Program hasznosulása

Fejlesztésünkkel egy olyan rendszert hoztunk létre, amely a diákok és a diákkormányzati tagok közötti kommunikációt nagymértékben megkönnyíti. A program lehetőséget ad események, bejegyzések, szavazások és fájlok egyszerű megosztására, valamint hozzájárulások, visszajelzések, és egymás közti kommunikáció lehetőségére. Törekedtünk arra, hogy a platform minden felhasználói csoport számára átlátható, gyorsan megtanulható és eszközfüggetlen legyen.

Hisszük, hogy fejlesztésünkkel egy valódi igényt sikerült lefednünk, amely hozzájárul az iskolai közösségi élet átláthatóságához, szervezettségéhez és hatékonyságához.

Köszönetnyilvánítás

Ezúton szeretnénk köszönetet mondani mindeneknek, akik közvetlenül vagy közvetve hozzájárultak munkánk megvalósításához.

Szeretnénk elsősorban köszönetet mondani tanárainknak, nem csak amiért az elmúlt években tanítottak és segítettek bennünket, hanem azért is, mert támogattak a feladat végbemenetelével, egyes megoldások kivitelezésének javaslásával, illetve lehetőséget adtak a tanórákon történő fejlesztésre. Támogatásuk nélkül nem juthattunk volna el idáig.

Hálásak vagyunk azoknak a tanároknak és iskolatársaknak is, akik tesztelésükkel, visszajelzésekkel és ötleteikkel hozzájárultak a program működésének tökéletesítéséhez.

Külön köszönetet szeretnénk nyilvánítani Kovács László Tanár úrnak, akinek köszönhetjük a projekt létrejöttét, és a folyamatos támogatását is.

Végül, de nem utolsósorban szeretnénk megköszönni családjaink türelmét és támogatását, akik a háttérben végig mellettünk álltak, és biztattak bennünket a projekt során.

Irodalomjegyzék

<https://infojegyzet.hu>

<https://www.php.net>

<https://stackoverflow.com>

<http://weissesvagyok.hu>

<https://www.w3schools.com>