



ZÁRÓDOLGOZAT

Készítették:

Lengyel Bálint

Kelemen Ádám

Juhász Balázs

Konzulens:

Farkas Zoltán

Miskolc

2024.

Miskolci SZC Kandó Kálmán Informatikai Technikum

Miskolci Szakképzési Centrum

SZOFTVERFEJLESZTŐ- ÉS TESZTELŐ SZAK

ZÁRÓDOLGOZAT

Connect2Gether (Közösségi média) felület

Lengyel Bálint - Kelemen Ádám - Juhász Balázs

2023-2024

TARTALOMJEGYZÉK

A PROJEKTRŐL.....	5
FELHASZNÁLT TECHNOLOGIÁK ÉS PROGRAMOZÁSI NYELVEK.....	6
FRONTEND.....	6
HTML:	6
CSS:	6
React:	7
Bootstrap:.....	7
BACKEND	8
Asp.NET Core 8.0 Web API:.....	8
Entity Framework Core:	8
XAMPP:	9
MYSQL:	9
JWT Token:	9
FELHASZNÁLT PROGRAMOK.....	10
Visual Studio 2022:.....	10
Visual Studio Code:.....	11
KOMMUNIKÁCIÓS FELÜLETEK.....	11
Trello:	11
Github:	11
Discord:	12
ADATBÁZIS SZERKEZET	13
WEBOLDAL BEMUTATÁSA	16
Főoldal:.....	16
Regisztráció:.....	16
Bejelentkezés:.....	17
Publikus profil oldal:.....	18
Post oldal:	19
Profil oldal:	19
Admin oldal:	19
Admin felhasználó profil oldal:	19
Admin irányító felület:.....	20
ASZTALI ALKALMAZÁS.....	21
WPF: (Windows Presentation Foundation).....	21

KEZELŐFELÜLET	21
Bejelentkezés:.....	21
Admin Home:.....	22
FORRÁSJEGYZÉK	26

A PROJEKTRŐL

A csoportunk egy közösségi média felületet készített webes és egy hozzátartozó WPF-es grafikus oldallal.

Sokat agyaltunk azon, hogy milyen projektet készítsünk, de végül azért döntöttünk e mellett, mivel kellően egyedinek éreztük ezt a témakört és izgalmasnak tartottuk, hogy egy konkrét közösségi média felületet készíthetünk. A mai világban szinte mindenki használ ilyen oldalakat, vegyük csak például a Facebook-ot, Twitter-t vagy akár az Instagramot. Mindegyikben az a közös, hogy az emberek megtudják osztani tartalmaikat a nyilvánossággal. Többféle formájuk létezik, mint például a szöveges tartalmak, a képek vagy a videók publikálása. Mi ezeken belül a szövegeset céloztuk meg. A felhasználó létrehoz egy postot és erre a többi felhasználó kommentelni tud. A projekt részleteit ebben a dokumentációban fogjuk elmagyarázni, hogy mindenkinek egyértelmű legyen a működése és részletes betekintést engedjünk a projekt fejlesztésébe.



1. ábra - A projekt logója

FELHASZNÁLT TECHNOLÓGIÁK ÉS PROGRAMOZÁSI NYELVEK

FRONTEND

HTML:

A HTML egy jelölő nyelv, kódnyelv, a weboldalak készítéséhez fejlesztették. Maga a rövidítés, az angol HyperText Markup Language (hiperszöveges jelölőnyelv) kifejezésből alakult mozaikszó.

A HTML a weboldalak létrehozásának hiper szöveges jelölő nyelve, amely leírja a weboldalunk felépítését. A hipertext, a világhálón lévő dokumentumok sokasága, amelyben képek, szövegek, animációk, videók, hanganyagok találhatóak.

A HTML formáját tekintve egy sorozat, de több elemet tartalmaz. Ezek az elemek mutatják meg a böngészőnek, a tartalom megjelenítését, egy dokumentumban, weboldalon: pl. dőlt betűvel, vastagon szedve stb. Mögötte egy-egy jelölés áll, amely nélkül csak formátlan, ömlesztett szöveg lenne.

CSS:

A CSS az angol (Cascading style sheets) kifejezés rövidítése, ami magyarul „egymásba ágyazott stíluslapokat” jelent. Míg a HTML a weblap szerkezetét határozza meg (főcímek, bekezdések stb.), és lehetővé teszi, hogy különböző elemeket ágyazz webes dokumentumban, addig a CSS a weblap vizuális stílusáért felel. Az oldal elrendezéséért, a színekért, a betűkészletekért, azok méretéért, és így tovább.

React:

A React.js, gyakran egyszerűen csak React-nak nevezik, egy Facebook által fejlesztett nyílt forráskódú JavaScript könyvtár, amely dinamikus felhasználói felületek létrehozására szolgál. Lehetővé teszi a fejlesztők számára, hogy olyan webalkalmazásokat hozzanak létre, amelyek hatékonyan frissíthetők és renderelhetők az adatváltozásokra válaszul, így ideális az egyoldalas alkalmazások fejlesztéséhez, ahol a sima felhasználói élmény kiemelt szerepet játszik.

A React komponens alapú architektúrájú ösztönzi a használható felhasználói felület komponensek létrehozását, ami javítja a kód konzisztenciáját és karbantarthatóságát.

A React.js az egyszerűsége, rugalmassága és skálázhatósága miatt a frontend fejlesztés egyik legnépszerűbb JavaScript könyvtárává vált. A képessége, hogy javítsa a webalkalmazások fejlesztési folyamatát és teljesítményét, a világ minden tájáról fejlesztők számára az egyik legjobb választássá tette.

Bootstrap:

A Bootstrap egy nyílt forráskódú keretrendszer (framework), mely HTML, CSS, JavaScript technológiákat használ. Alapvetően arra jó, hogy nagyon könnyedén, és minimális energia befektetéssel tudjon valaki jól kinéző, bármilyen képernyőméreten szépen megjelenő weboldalakat készíteni.

A Bootstrap előnye, hogy a legtöbb „problémára”, mellyel egy fejlesztő találkozik weboldalak arculatának kialakításakor, már egy előre megoldást kínál, ami vagy teljesen megfelel az adott célra, vagy pedig minimális átalakítással.

Axios:

Az Axios egy egyszerű ígért alapú HTTP-kliens a böngésző és a node.js számára. Az Axios egy egyszerűen használható könyvtár biztosít egy kis csomagban, nagyon bővíthető felülettel.

BACKEND

Asp.NET Core 8.0 Web API:

Az ASP.NET Web API egy kiterjeszthető keretrendszer HTTP alapú szolgáltatások építésére, amelyekhez különböző platformokon, például weben, Windowson, mobil eszközökön keresztül lehet hozzáférni. Az ASP.NET MVC webalkalmazáshoz hasonlóan működik, kivéve, hogy adatokat küld válaszként, ahelyett, hogy HTML nézetet küldene. Ez olyan, mint egy webszolgáltatás vagy WCF szolgáltatás, de az kivétel, hogy csak az HTTP protokollt támogatja.

Entity Framework Core:

Az Entity Framework Core egy nyílt forráskódú, könnyű súlyú, cross-platform ORM (Object-Relational Mapping) keretrendszer. Lehetővé teszi az alkalmazás fejlesztők számára, hogy a .NET alkalmazásaikat objektum-orientált módon építsék fel.

Az Entity Framework Core támogatja a különböző adatbázis-motorokat, például az SQL Server és a MySQL. Lehetőség van az adatbázisokkal való kommunikációra a LINQ (Language Integrated Query) nyelv használatával.

Könnyen integrálható az ASP.NET Core alkalmazásokban, és számos szolgáltatást és lehetőséget kínál az adatbázis-kezeléssel kapcsolatban, amelyek

nagyban segítik az alkalmazás fejlesztőket az adatbázisok hatékony és biztonságos kezelésében.

XAMPP:

A XAMPP egy ingyenes és nyílt forráskódú szoftvercsomag, amelyet webfejlesztők használnak webkiszolgálók létrehozásához és üzemeltetéséhez.

MYSQL:

A MySQL egy nyílt forráskódú relációs adatbázis-kezelő rendszer, amely lehetővé teszi az adatok hatékony kezelését és tárolását. Egyik előnye az adatbázis-skálázhatóság, ami azt jelenti, hogy a rendszer képes megbírkózni nagy mennyiségű adattal és sok egyidejű felhasználóval.

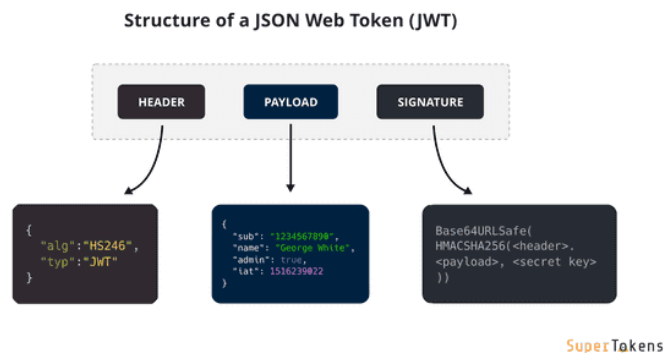
Az egyik legnépszerűbb relációs adatbázis-kezelő rendszer a világon, és széles körben használják a különféle alkalmazásokban, például webalkalmazásokban, adatbázisokban, szoftverekben és még sok másban. Az adatok lekérdezése SQL nyelven keresztül történik, amely az adatbázis-kezelő rendszerek leggyakoribb nyelve.

JWT Token:

A JSON Web Token (JWT) egy nyitott szabvány, amely meghatározza a felek közötti információk biztonságos továbbításának kompakt és önálló módját JSON objektumként. Ez az információ ellenőrizhető és megbízható, mivel digitálisan alávan írva. A JWT-ket titok felhasználásával lehet aláírni vagy nyilvános / magán kulcspár felhasználásával RSA vagy ECDSA.

Kompakt formájában a JSON Web Tokens három részből áll: első a fejléc (header) jellemzően két részből áll: a token típusa, amely JWT, és az alkalmazott

aláíró algoritmus, például HMAC SHA256 vagy RSA. A második a tartalom (payload) ez tartalmazza a követeléseket ezek egy entitásra (általában felhasználóra) vonatkozó állítások és kiegészítő adatok, háromféle követelés létezik: bejegyzett, nyilvános és magán igények. A harmadik pedig az aláírás (signature), annak ellenőrzésére szolgál, hogy az üzenet nem változott-e az út során, és magánkulccsal aláírt tokenek esetében, azt is ellenőrizheti, hogy a JWT feladója az, aki azt mondja.



2. ábra - JWT Token működése

FELHASZNÁLT PROGRAMOK

Visual Studio 2022:

A Visual Studio egy hatékony fejlesztői eszköz, amellyel a teljes fejlesztési ciklust egy helyen végezheti el. Ez egy átfogó integrált fejlesztői környezet (IDE), amellyel kódot írhat, szerkeszthet, hibát kereshet és kódot építhet, majd telepítheti az alkalmazást.

A kódszerkesztésen és hibakeresésen túl a Visual Studio fordítót, kódkiegészítő eszközöket, forrásvezérlést, bővítményeket és még sok más funkciót tartalmaz a szoftverfejlesztési folyamat minden szakaszának javításához.

A Visual Studio számos funkciójának és nyelvi támogatásának köszönhetően az első „Hello World” program megírásától az alkalmazások fejlesztéséig és üzembe helyezéséig fejlődhet.

Visual Studio Code:

A Visual Studio Code (röviden VS Code vagy Code) a Microsoft cég által fejlesztett ingyenes és nyílt forráskódú szövegszerkesztő. Támogatja az automatikus kódkiegészítést (IntelliSense, Emmet, TabNine), rendelkezik beépített hibakeresővel, és Git verziókezelő támogatása, és még számos egyéb szolgáltatással. Bővítmények segítségével további szolgáltatások, és funkciók adhatóak hozzá.

A felhasználói igényeknek megfelelően testre szabható, ebben sokat segítenek a letölthető színsémák. A VS Code több platformos alkalmazás, elérhető Windows, Linux és OS X operációs rendszerekre.

KOMMUNIKÁCIÓS FELÜLETEK

Trello:

A Trello olyan vizuális eszköz, amely segíti a csapatok bármilyen projekt-, munkafolyamat- vagy feladatkövetés kezelésében. Hozzálehet adni fájlokat, ellenőrzőlistákat, vagy akár automatizálást: az csapatok úgy szabhatják testre, ahogy nekik a legjobb.

Github:

A GitHub egy felhőalapú platform, amely a Gitet, egy elosztott verziókövetési rendszert használ a magjában. A GitHub platform leegyszerűsíti a

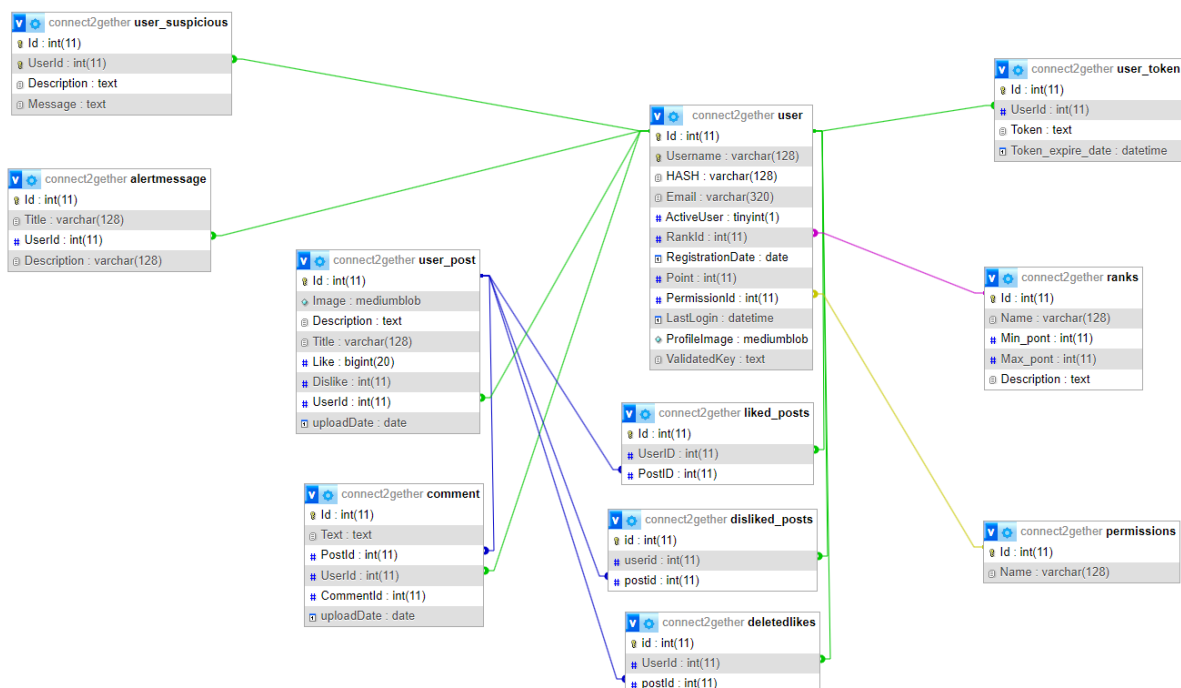
projekteken való együttműködés folyamatát, és egy webhelyet, parancssori eszközöket és általános folyamatot biztosít, amely lehetővé a fejlesztők és a felhasználók együttműködését.

Discord:

A Discord, egy VoIP (Voice-over-Internet Protocol) alkalmazás, ami lehetőséget nyújt a kommunikációra a felhasználók között. A készítőket azt a szempontot vették figyelembe a tervezés során, hogy a nagy gépigényű alkalmazások futtatásával egyidejűleg működőképes alkalmazást hozzanak létre, úgy, hogy az a gép erőforrásait ne használja feleslegesen.

A funkciói miatt számos területen helyt tud állni, akár a mi projektünk készítése közben fontos kommunikációs platformként üzemel. Könnyű kezelhetősége és funkciói megadják a lehetőséget, hogy létrehozzanak külön szobákat egyes projektekhez vagy az eszközök kijelzőinek streamelésével bemutathassák egymásnak munkáikat.

ADATBÁZIS SZERKEZET



3. ábra - Adatbázis szerkezete

A **user** táblában tároljuk a felhasználó adatait, az id-ját (ami az elsődleges kulcs), a felhasználó nevét, a hashelt jelszót, az email címet, az **ActiveUser** ami kezdetben 0 azaz false értékkel tér vissza a regisztráció után. Miután validálják a felhasználót ez az érték 1 azaz true-val tér vissza és utána be lehet jelentkezni ezzel a felhasználóval. A **rankid**-t (ami egy idegen kulcs), a **RegistrationDate**-t ami a regisztráció dátuma, a **point** ami az adott felhasználó pontszáma, a **permissionid** (ami szintén egy idegen kulcs), a **lastlogin** ami a legutolsó bejelentkezés dátumával tér vissza, a **profileimage** a felhasználó profilképe és a **validatedkey** ami a profil validálásához szükséges.

A **user_post** táblában tároljuk a postok adatait, az id-ját (ami az elsődleges kulcs), az **image** ami a poston található kép amennyiben a posthoz lett adva kép. A **description** a post szövege, a **title** a post címe, a **like** és **dislike**, a **userid** (ami

egy idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, akié a post és az uploaddate ami a post feltöltésének dátuma.

A **comment** táblában tároljuk a kommentek adatait, az id-ját (ami az elsődleges kulcs), a text a komment szövege, a postid (ami egy idegen kulcs, ez van összekötve a user_post tábla id-jával) az a post, ahová írva lett a komment, a userid (ami szintén idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, akié a komment, a commentid hogyha egy kommentre kommentelnek annak az id-ja és az uploaddate ami a komment feltöltésének dátuma.

Az **alertmessage** táblában tároljuk a felhasználó figyelmeztető üzeneteinek adatait, az id-ját (ami az elsődleges kulcs), a title ami az üzenet címe, a userid (ami szintén idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, aki kapta az üzenetet és a description a komment szövege.

A **ranks** táblában tároljuk a rangok adatait, az id-ját (ami az elsődleges kulcs, ez van összekötve a user tábla rankid-jával), a rang nevét, a rang minimum és maximum pontszáma és a description ami a rang leírása.

A **permissions** táblában tároljuk a role-ok id-ját (ami az elsődleges kulcs, ez van összekötve a user tábla permissionid-jával) és a role nevét.

A **user_token** táblában tároljuk a tokennek id-ját (ami az elsődleges kulcs), a userid (ami szintén idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, akié a token, maga token és a token lejárat dátuma.

A **user_suspicious** táblában tároljuk a gyanús felhasználók userid-ját (ami az elsődleges kulcs, ez van összekötve a user tábla id-jával), a description-jét és message-ét.

A **liked_posts** táblában tároljuk a userid-t (ami egy idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, aki likeolt és a postid-t (ami

szintén egy idegen kulcs, ez van összekötve a user_post tábla id-jával) az a post, ami likeolva lett.

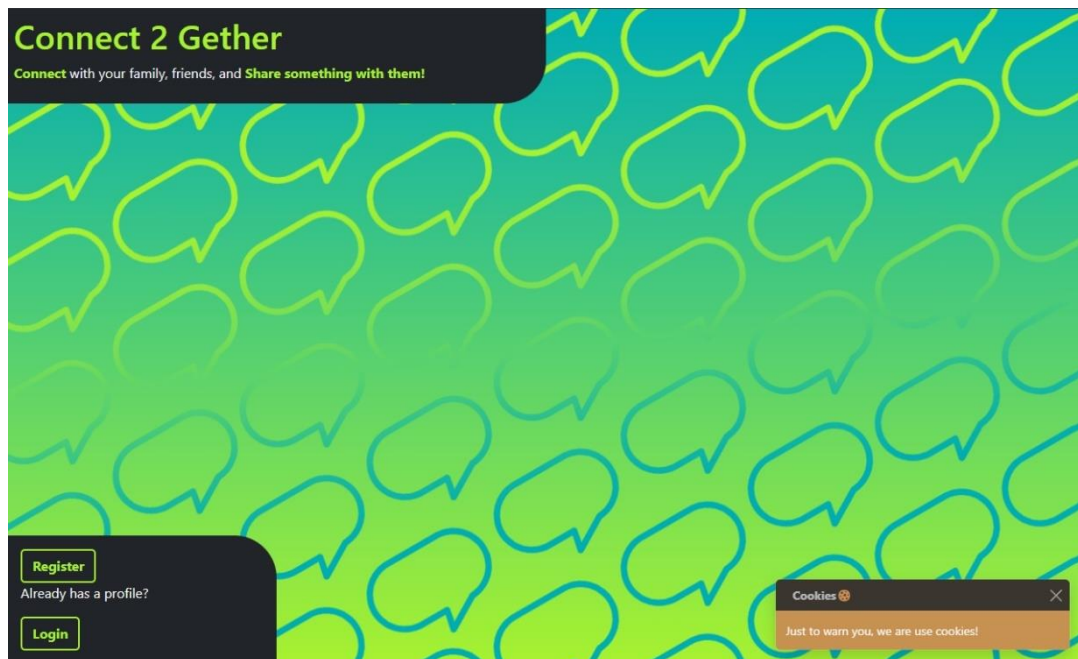
A **disliked_posts** táblában tároljuk a userid-t (ami egy idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, aki dislikeolt és a postid-t (ami szintén egy idegen kulcs, ez van összekötve a user_post tábla id-jával) az a post, ami dislikeolva lett.

A **deletedlikes** táblában tároljuk a userid-t (ami egy idegen kulcs, ez van összekötve a user tábla id-jával) az a felhasználó, aki már likeolt, kapott pontot és hogyha újra likeolja ugyanazt a postot akkor már nem kap pontot, és a postid-t (ami szintén egy idegen kulcs, ez van összekötve a user_post tábla id-jával) az a post, ami már likeolva volt.

WEBOLDAL BEMUTATÁSA

Főoldal:

A főoldalon a cím alatt található egy rövid leírás az oldal funkciójáról vagy rendeltetéséről. Az oldal alján találjuk a bejelentkezést és regisztrációt.



4. ábra - A főoldal

Regisztráció:

A regisztrációnál meg kell adjunk egy Felhasználó nevet, egy email címet és egy jelszót. A jelszónak tartalmaznia kell kis és nagy betűket, legalább egy darab számot, egy speciális karaktert és legalább 8 karakter hosszúságúnak kell lennie. Regisztrálás után az email címre érkezik egy hitelesítő link, amit meg kell nyitni, ami átirányít egy oldalra, ahol egy gombnyomással hitelesíthetjük a fiókot. Ez után már bejelentkezhethetünk.

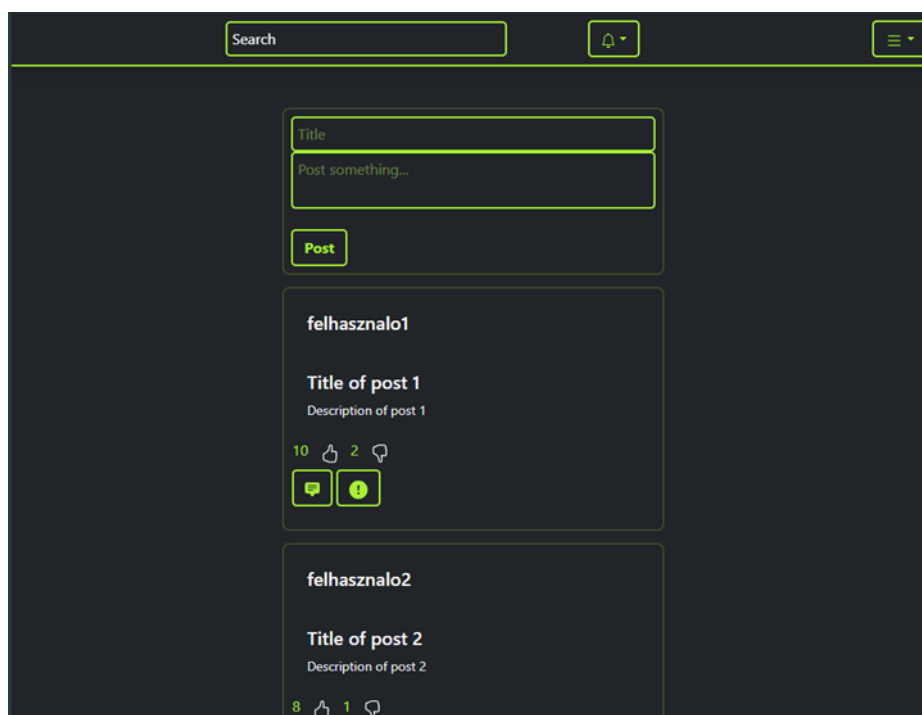
Bejelentkezés:

A bejelentkezésnél meg kell adjuk a felhasználó nevet és jelszavunkat. Amennyiben sikeres a bejelentkezés akkor átirányít a felhasználói főoldalra.

A „Forgot your password?” felíratra kattintva módosíthatjuk jelszavunkat ha elfelejtettük volna azt. A szövegre kattintás után meg kell adjuk az profilhoz tartozó email címet. Az oldal küld egy linket amire kattintva átirányít egy oldalra, ahol megadhatjuk az új jelszavunkat.

Felhasználói oldal:

Ez az oldal fogad bejelentkezés után. Itt érjük el az összes posztot és itt tudjuk azokat likeolni vagy dislikeolni, ha postnál egy gombbal meg tudjuk jeleníteni a kommenteket és itt mi is tudunk írni azt. Ha a post szövegére kattintunk akkor át irányít egy oldalra, ahol csak azt a postot láthatjuk. Ha a post feltöltőjének vagy az egyik kommentelőnek a nevére kattintunk akkor pedig a profilját tudjuk megtekinteni.



5. ábra - Felhasználói oldal

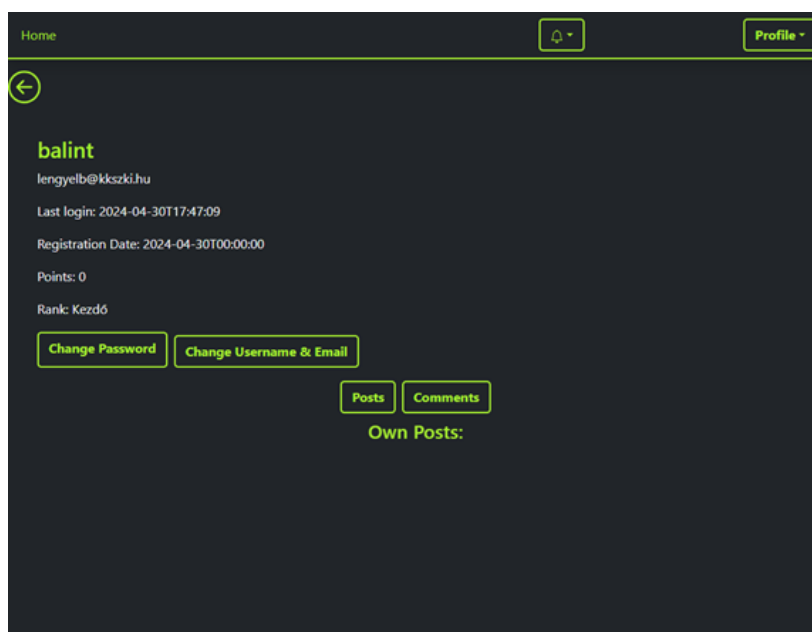
A postok felett található az új post feltöltése, aminél meg kell adni egy címet és egy szöveget.

A navigációs panelen érjük el a keresőt, amiben felhasználót és postot is kereshetünk, Ha csak szimplán beleírunk akkor a post címét keresi, ha egy @ jelet írunk a szó elé akkor pedig a felhasználó nevét keresi. Ha @ jellel beírunk egy felhasználó nevet és utána szóközzel elválasztva elkezdünk egy post címet írni akkor az attól a felhasználótól származó posztokba keresi a címet.

A Navigációs panelen érjük el egy gomb segítségével a további lehetőségeket. Itt tudjuk az oldal színét átváltani világosról sötétre vagy fordítva. A „Details” gombbal pedig a profilunk adataihoz navigál.

Publikus profil oldal:

Ez az oldal akkor jelenik meg ha egy felhasználó nevére kattintunk a postnál vagy kommentnél. Itt jelennek meg azok az adatok, amik publikusak, mint a felhasználó név, rang, pontszám, profil létrehozásának a dátuma, és az utolsó bejelentkezés ideje. A felhasználó postjait is láthatjuk itt.



6. ábra - Publikus profil oldal

Post oldal:

Ez az oldal akkor jelenik meg ha egy post szövegére kattintunk. Itt azt az egy postot jeleníti meg az oldal a hozzászólásokkal. Ennek az oldalnak az url -ét kimásolva megoszthatunk egy postot.

Profil oldal:

A navigációs panelen a „Profile” gombnál a „Details” fülre kattintva érjük el ezt az oldalt. Itt érjük el a személyes adatainkat, mint például az email címünket. A jelszavunkat itt változtathatjuk meg a régi jelszó megadásával. A felhasználó nevünket és email címünket is megváltoztathatjuk itt.

A saját postjainkat és kommenteket is láthatjuk, amiket törölhetünk és módosíthatunk is innen.

Admin oldal:

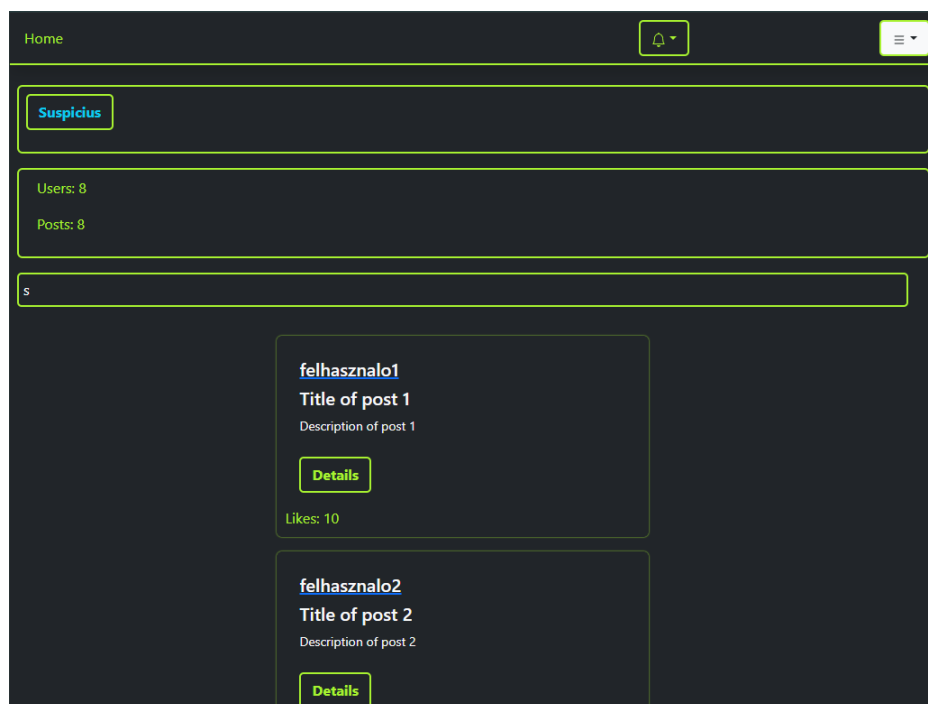
Amikor admin fiókkal jelentkezünk be, egy hasonló oldalra irányít, mint egy felhasználó, itt viszont több funkciót érünk el. A postoknál megjelenik egy felkiáltójeles gomb, amivel lehet üzenetet küldeni a post feltöltőjének. Ezt az üzenetet az oldalon is megkapja, amit a navigációspanelen a csengőnél tud megnézni, és email formájában is megkapja.

Admin felhasználó profil oldal:

Ha egy Felhasználó profilját megnyitjuk akkor több adatot láthatunk róla, mint amit egy felhasználó látna. Felhasználót itt megjelölhetjük gyanús felhasználóként módosíthatjuk az adatait vagy törölhetjük a felhasználót.

Admin irányító felület:

A navigációs felületen, ha lenyitjuk a jobb szélén található menüt akkor érhetjük el az „Admin Dashboard” -ot. Ezen a felületen láthatjuk egy listába a gyanús felhasználókat. Látjuk hány darab felhasználó és post van. Valamint rá tudunk keresni felhasználókra és postokra is. Itt is tudjuk törölni vagy módosítani az értékeket.



7. ábra - Admin irányító felület

ASZTALI ALKALMAZÁS

WPF: (Windows Presentation Foundation)

A Windows Presentation Foundation (WPF) egy fejlesztési keretrendszer, amelyet asztali alkalmazások létrehozására használnak. Ez a .NET keretrendszer része. A WPF-nek felbontásfüggetlen és vektoralapú renderelő motorja van, ami segít a modern grafikus hardverekkel való kezelésben. A legfrissebb verziója a 8.0.0. Ebben a keretrendszerben az alkalmazás felhasználói felületét XAML nyelven tervezik, míg az alkalmazás logikáját a C# programozási nyelven írják.

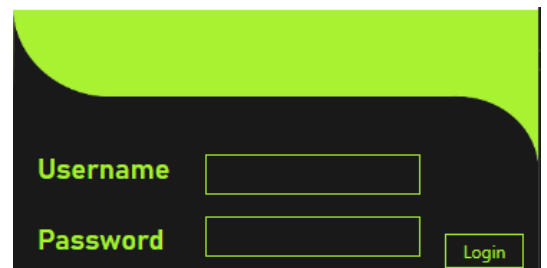
KEZELŐFELÜLET

Bejelentkezés:

Az alkalmazás megnyitásakor a bejelentkezői felület jelenik meg, ahol a felhasználó egy Admin vagy Moderátor jogosultsággal rendelkező fiókkal tud belépni. A Teszthez használható.

Admin fiók:

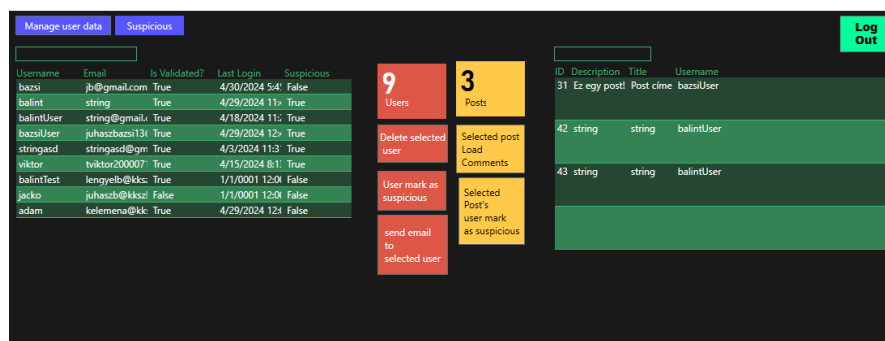
- balint
- Asdasd123+



8. ábra: Bejelentkezés ablak

Admin Home:

A bejelentkezést követően, az „Admin Home” felület jelenik meg, ahol az Adminnak lehetősége adódik a felhasználók adatainak kezelésére.



Az oldalon két táblázat (DataGrid) található, a bal oldalon a felhasználókkal a jobb oldalon pedig a felhasználók posztjai találhatóak meg. Mindkét helyen található keresőmező, ahol felhasználó nevekre, és e-mailekre és egyéb más adatokra tudunk szűrni, megkönnyítve a kezelőfelület használatát.

9. ábra: Admin Home felület

Középen találhatóak a Kezelőfelület gombjai, és a számlálók a felhasználók, illetve a posztok számával. A felhasználók kezelésére három gomb szolgál:

1. Delete selected user:

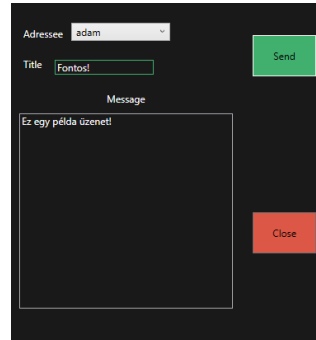
Ez a gomb a kiválasztott felhasználótörlésére szolgál.

2. Mark as suspicious:

Ez a gomb arra szolgál, ha felhasználói tevékenység megkérdőjelezhető a Moderátorok vagy az Adminok által akkor megtudják őket jelölni későbbi felülvizsgálatra.

3. Send email to selected user:

Ez a gomb közvetlen email küldési lehetőséget biztosít a választómenüben kijelölt felhasználónak, a bejelentkezett Admin vagy Moderátori fiók nevével.



11. ábra: E-mail küldési felület

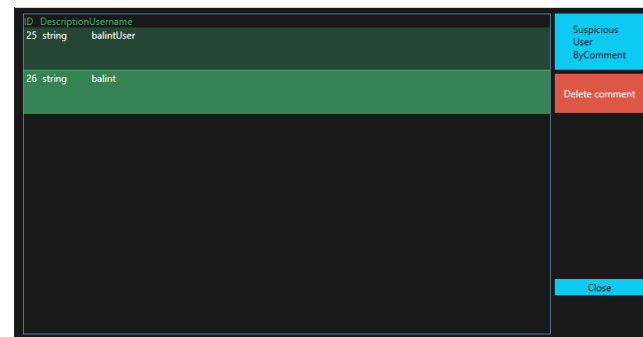


10. ábra: Admin e-mailes értesítése a felhasználónak

A posztok kezelésére két gomb szolgál:

1. Selected post load comments:

Ezzel a gombbal lehetőséget kapunk a poszthoz tartozó kommentek megjelenítésére, itt láthatjuk a poszthoz tartozó kommenteket, és az adataikat (szerző, tartalom). Itt lehetőség adódik a felhasználót megjelölni gyanús tartalom miatt, illetve a komment törlésére.



12. ábra: Poszthoz tartozó komment kezelési felület

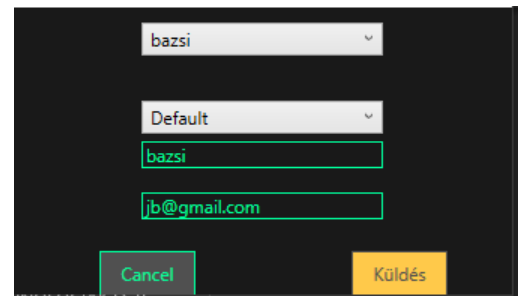
2. Selected Post's user mark as suspicious:

Ezzel a gombbal megjelölheti az alkalmazás használója a felhasználót, ha megkérdőjelezhető posztot tett ki.

Az Admin Home bal felső sarkában két gomb található:

1. Manage user data:

Itt lehetőség adódik a felhasználó adatainak módosítására. A fenti legördülő menüben kiválaszthatjuk a felhasználót, és ezáltal a lenti menüben automatikusan betöltődnek az adatai, itt módosíthatjuk a jogosultságát, nevét, illetve az e-mail címét, erről pedig emailben értesítjük az érintett felhasználót, ahol megkapja a módosított adatait a profiljához immár az új e-mail címére.



8. ábra: Felhasználó adat módosítása



9. ábra: Adat módosításról kapott értesítés

2. Suspicious:

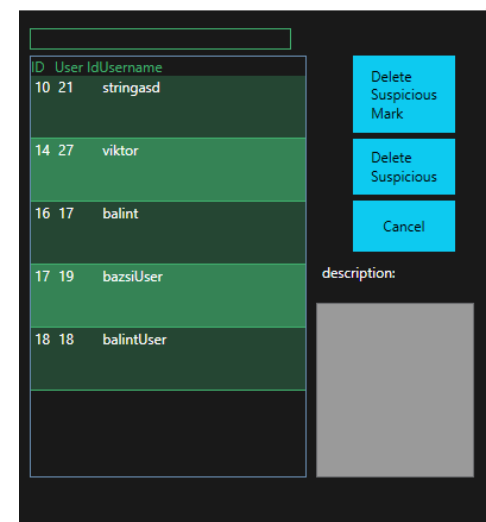
Ezzel a gombbal megjeleníthetjük a „gyanús” -ként megjelölt felhasználókat egy külön menüben.

Itt szintén található egy lista felette egy keresőmenüvel, amivel böngészhetünk a „gyanús” felhasználók között, felhasználó ID és felhasználó név alapján.

a menü jobb oldalán három gomb lett elhelyezve:

1. Delete Suspicious Mark:

Ha felhasználót jogtalanul jelöltük meg akkor ennek a gombnak a segítségével törölhetjük a „gyanús” felhasználók listájából



10. ábra: Gyanús felhasználók kezelőfelülete

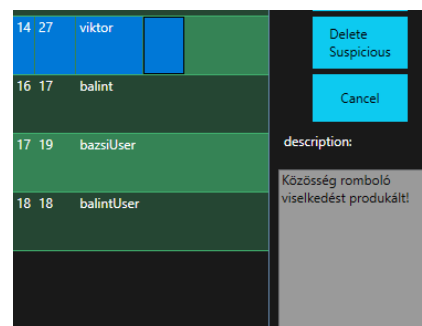
2. Delete Suspicious:

Ezzel a gombbal törölhetjük a kijelölt „gyanús” felhasználót véglegesen, ha véglegesen úgy döntöttünk, hogy rontja weboldalunk közösségét.

3. Cancel:

Ezzel a gombbal visszaléphetünk az Admin Home menübe.

A gombok alatt lévő mező a „gyanús” jelölés okát jeleníti meg amikor kijelöljük a felhasználót, ez egy Admin által megadott indok, arra, hogy miért kapott jelölést a felhasználó.



Az Admin az „Admin Home” jobb felső sarkában található „Log Out” gombbal tud kijelentkezni a felületről, ami visszairányítja őt a „Login” nevezetű bejelentkezési felületre.

11. ábra: Gyanús felhasználó leírása

FORRÁSJEGYZÉK

1. Téma: HTML, webcím: <https://hwellkft.hu/marketing-szotar/html>
2. Téma: CSS, webcím: <https://matebalazs.hu/css.html>
3. Téma: JWT Token, webcím: <https://jwt.io/introduction>
4. Téma: React, webcím: <https://loadfocus.com/hu-hu/glossary/react-js>
5. Téma: Bootstrap, webcím: <https://gremmedia.hu/mi-az-a-bootstrap-4-hogyan-hasznaljuk>
6. „1.ábra”: JWT Token, webcím: <https://supertokens.com/blog/what-is-jwt>
letöltés dátuma: 2024.04.27
7. Téma: Visual Studio Code, webcím: <https://digikiad.gitbook.io/digitalis-kiadvanyok/vs-code/mi-a-vs-code>
8. Téma: MySQL, webcím: <https://www.oracle.com/mysql/what-is-mysql/>
9. Téma: Asp .Net Web API, webcím: <https://www.tutorialsteacher.com/webapi/what-is-web-api>