

VIZSGAREMEK

Hegyesi Balázs Tibor, Bakk Noel Csaba, Marosvölgyi Martin György

Budapest, 2025

BGÉSZC SZILY KÁLMÁN TECHNIKUM ÉS KOLLÉGIUM

SZOFTVERFEJLESZTŐ ÉS -TESZTELŐ

13.A

HISTÓRIA TÖRTÉNELMI WEBOLDAL

Konzulens:

Lipák Tibor

Készítette:

**Hegyesi Balázs Tibor, Bakk Noel Csaba,
Marosvölgyi Martin György**

BUDAPEST

2025

Tartalom

1	Bevezetés	6
1.1	A weboldal funkciója	6
1.2	Miért érdemes regisztrálni és bejelentkezni?	6
1.3	Az oldal jövője	7
2	Felhasználói felület bemutatása	8
2.1	Kezdőlap	8
2.2	Rólunk – avagy a Históriaokról	10
2.3	„Kisoldalak”	11
3	Regisztráció és bejelentkezés	14
3.1	Regisztráció	14
3.2	Hitelesítés	15
3.3	Bejelentkezés	16
3.4	Regisztráció és bejelentkezés ablak (modal) stílusa	17
3.5	Jelszóerősség működése:	17
4	Adatbázis	19
4.1	Az adatbázis bemutatása	19
4.2	Adatbázis szerkezete	19
4.3	Táblák részletes leírása	20
4.4	Tervezése	21
4.5	Előnyök	22
5	Backend fő funkciói	23
5.1	Adatbázis Kezelés	23
5.2	Regisztráció és Email Hitelesítés	23
5.3	Bejelentkezés és Munkamenet Kezelés	25
5.4	Hibakezelés:	27
6	Backend Fejlesztői Dokumentáció	29
6.1	Fájlok és Funkciók	29
6.2	Hibakezelés	29
6.3	Biztonság	30
7	Fejlesztés	31
7.1	Frontend	31
7.2	Backend	31

7.3	Adatbázis	31
7.4	Fejlesztési nehézségek	32
7.5	Tanulságok	32
8	<i>Tesztelési dokumentáció</i>	33
9	<i>Rendszerkövetelmények</i>	35
9.1	Minimum rendszerkövetelmények	35
9.2	Ajánlott rendszerkövetelmények	35
<i>Források és használt programok</i>		36
Források		36
Használt programok		36

Ábrajegyzék

1. ábra főoldal.....	8
2. ábra idővonal	9
3. ábra szocial oldalak	9
4. ábra	10
5. ábra kisoldal	11
6. ábra regisztráció	14
7. ábra hitelesítés	15
8. ábra bejelentkezés	16
9. ábra jelszóerősség mérő	18
10. ábra adatbázis	20

1 BEVEZETÉS

Vizsgaremek projektünk témájául a 20. század egyetemes és magyar történelmének bemutatását választottuk. Fontosnak tartjuk ezt a korszakot, mivel eseményei – a világháborúktól kezdve a rendszerváltásig – máig meghatározzák a világ és hazánk társadalmi, politikai és gazdasági viszonyait. Célunk egy olyan weboldal létrehozása volt, amely korszerű módon, interaktív elemek segítségével segíti a felhasználókat a történelmi események megismerésében és rendszerezésében.

1.1 A weboldal funkciója

A weboldal egyik fő különlegessége az idővonal, amelyek lehetőséget nyújtanak a felhasználóknak arra, hogy térben és időben is áttekinthetően navigáljanak a történelem főbb eseményei között, legyen szó magyar, vagy egyetemes történelemről. Az idővonalat gombok megnyomásával hívhatunk elő, „Magyar történelem” illetve „Egyetemes történelem” ezek gombok megnyomása után megjelenik/megjelennek az animált idővonal/idővonalak. Az idővonalon egy rövid leírást olvashatunk amely összefoglalja a témát. Amennyiben felkeltette érdeklődésünket amit olvastunk ebben a rövid részben a „Tovább olvasok”-ra kattintva elolvashatja kidolgozott témaköreinket, képekkel, és egy oldalon fel lelhető interaktív térkép segítségével.

1.2 Miért érdemes regisztrálni és bejelentkezni?

A bejelentkezés lehetőséget ad arra, hogy teljes mértékben kihasználhasd a História oldal nyújtotta funkciókat. Regisztrált felhasználóként elérhetővé válik számodra a „jegyzetkészítő” funkció, amellyel könnyedén feljegyezheted saját gondolataidat, fontos dátumokat vagy épp vizsgára való készülés során hasznos információkat. Jegyzeteid biztonságosan tárolódnak, és később bármikor visszatérhetsz hozzájuk – így az oldal nemcsak egy információforrás, hanem egy személyre szabott tanulási eszköz is lesz számodra. Azonban ez a funkció egyenlőre korlátozott oldalakon elérhető egyenlőre, későbbiekben az összes oldalon elérhetővé válik.

A fejlesztés során törekedtünk arra, hogy a weboldal reszponzív, esztétikus és felhasználóbarát legyen, külön figyelmet fordítva arra, hogy akár tanárok, akár diákok számára hasznos eszközként szolgáljon tanuláshoz vagy oktatáshoz. Projektünket úgy alakítottuk ki, hogy a tartalmak könnyen kereshetők és érthetően strukturáltak legyenek, ezzel támogatva a hatékony ismeretszerzést.

1.3 Az oldal jövője

A jövőben is szeretnénk tovább foglalkozni a weboldallal, folyamatosan bővítve a tartalmát, valamint kijavítva a jelenleg fennálló kisebb hibákat. Célunk, hogy a projektünket egy saját domain név alatt tegyük elérhetővé, ezzel biztosítva, hogy minél több érdeklődőhöz eljuthasson ez a digitális történelmi segédanyag, ezzel lehetővé téve a hosszas keresések mellőzését.

2 FELHASZNÁLÓI FELÜLET BEMUTATÁSA

2.1 Kezdőlap

Weboldalunk főoldala egy letisztult, történelmi hangulatot idéző felület, amely célja, hogy már az első pillanattól bevonja a látogatót a 20. század világába. (1. ábra) A központi szlogen – „A század, ami megváltoztatta a világot – ismerd meg a 20. századot!” – kiemelt helyen jelenik meg, ezzel megalapozva a tartalom hangulatát.



1. ábra főoldal

Az oldal kialakítása teljes mértékben reszponzív, vagyis minden eszközön – legyen az asztali gép, laptop, tablet vagy okostelefon – megfelelően jelenik meg és jól használható. A menüsor, a gombok, a háttérkép, valamint a szövegek és elrendezések automatikusan alkalmazkodnak a kijelző méretéhez, így biztosítva az optimális felhasználói élményt.

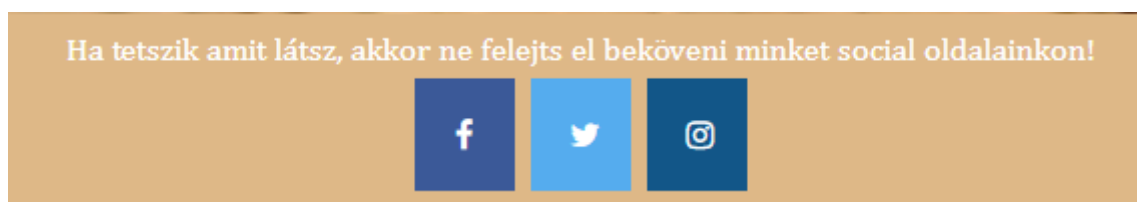
A kezdőlapon két fő kategória választható: Magyar történelem és Egyetemes történelem. A gombok egy-egy interaktív idővonalhoz vezetnek, ahol az adott témakörök legfontosabb eseményei szerepelnek (2. ábra). Az eseménypontokra kattintva rövid leírás jelenik meg, a „Tovább olvasok” gombra kattintva pedig részletes aloldalak nyílnak meg, amelyek mélyebb tartalmat, képeket, forrásokat és kiegészítő információkat is tartalmaznak.



2. ábra idővonal

Az idővonal megvalósítása HTML és CSS segítségével történt, külön JavaScript használata nélkül. A gördülékeny animációk és vizuális elemek támogatják a tanulást és az élményszerű felfedezést. Az oldal minden technikai és vizuális elemében arra törekedtünk, hogy az informatív tartalom modern, felhasználóbarát köntösben jelenjen meg.

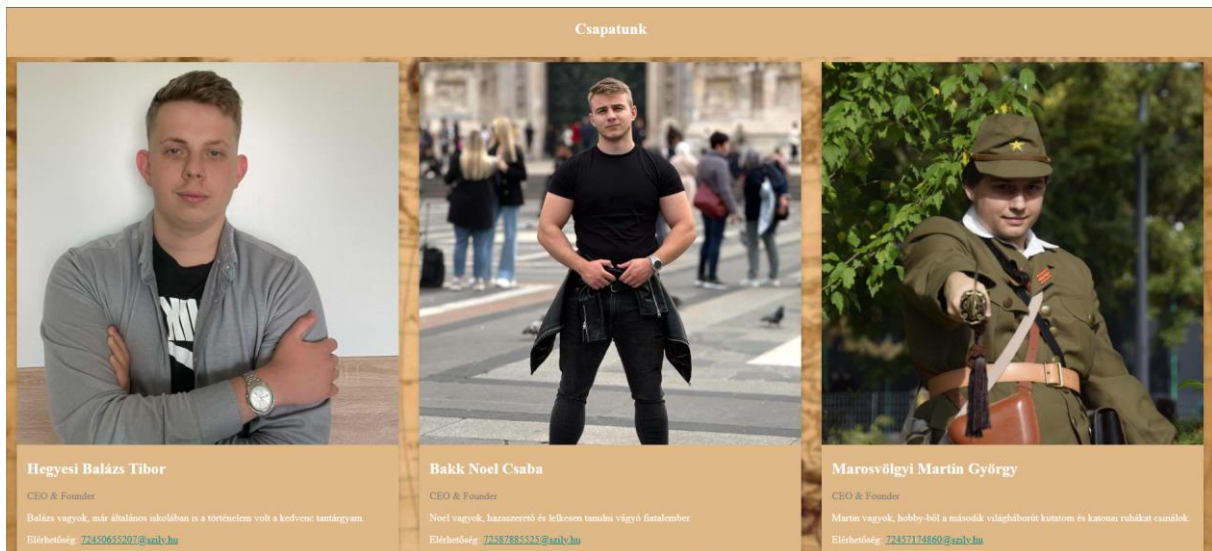
A weboldal alján elhelyeztünk egy footer szekciót, amely tartalmazza a projektünkhöz tartozó valóban működő közösségi média ikonokat (3. ábra). Ezek az ikonok saját, a projekt számára létrehozott Facebook, X/Twitter és Instagram oldalainkra mutatnak, amelyeken keresztül a felhasználók kapcsolatba léphetnek velünk, és naprakész információkat találhatnak a weboldal frissítéseiről és újdonságairól. Ezzel is arra törekedtünk, hogy a História ne csak egy passzív olvasási élményt nyújtson, hanem egy aktív, közösségi élményt is teremtsen.



3. ábra szocial oldalak

2.2 Rólunk – avagy a Históriásokról

A „Rólunk” oldal célja, hogy közelebb hozzuk magunkat a látogatókhoz, bemutatva, kik állnak a História projekt mögött (4. ábra). Ezen az oldalon rövid szöveges bemutatkozások és személyes fotók segítségével ismerhetik meg az oldal készítőit, illetve elérhetőségeinket is megtalálják – így lehetőséget biztosítunk arra, hogy a felhasználók kérdés, visszajelzés vagy észrevétel esetén közvetlenül kapcsolatba léphessenek velünk.



4. ábra

Az oldal tetején elhelyezett szöveg célja, hogy összefoglalja a História küldetését: közelebb hozni a 20. század eseményeit egy új, korszerű, vizuálisan is élményszerű formában. Kiemeltük, hogy célunk nem csupán az információátadás, hanem az élményszerű tanulás támogatása és a múlt megértésén keresztül való jelenbeli eligazodás.

A dizájn követi az oldal egységes arculatát: tört színek, klasszikus betűtípusok és letisztult, de informatív megjelenítés. A kontaktok kattinthatóak, így azonnal megnyílik az email kliens, ha valaki írni szeretne. Ez különösen hasznos, ha egy felhasználó hibát fedez fel, javaslata van vagy épp együttműködni szeretne velünk.

2.3 „Kisoldalak”

A kisoldalak a projekt alappilléret jelentik (5. ábra). Minden korszakhoz tartozik egy külön aloldal, amely vizuálisan és tartalmilag is részletesen bemutatja az adott időszak történéseit. Ezek az oldalak jól strukturáltak, tagolt szövegekkel, illusztrációkkal és forrásidézetekkel vannak ellátva, így nemcsak informatívak, hanem élményszerűek is.

A kisoldalak **célja**, hogy:

- Interaktívabbá és élvezhetőbbé tegyék a tanulást,
- a felhasználók ne csak lineárisan, hanem vizuálisan is tudják követni a történelmi eseményeket,
- átfogó képet adjanak egy adott korszakról,
- támogassák az önálló tanulást és a digitális oktatást.



5. ábra kisoldal

Technikai háttér és felépítés:

A kisoldalak HTML-ben készültek el, a tartalom strukturálásához és a különféle szakaszok elrendezéséhez ez az alap. A megjelenést CSS biztosítja, amelynek segítségével minden kisoldal saját háttérképet kapott.

A HTML kódon belül különféle section és div elemek vannak használva, amelyek a következő témákat fedik le:

- Előzmények
- Fontos évek és események (pl. 1914, 1915, stb.)
- Híres idézetek és történelmi források
- Illusztrációk és képgalériák
- Térképek, frontvonalak, csaták leírása

Reszponzivitás:

A teljes weboldal a későbbiekben mobilbarát megoldásokkal lesz kiegészítve, vagyis különféle képernyőméretekre is optimalizált. A képek és szövegek rugalmas elrendezésben fognak megjelenni, ami lehetővé teszi, hogy mobilon és asztali gépen is jól olvasható legyen.

Képi anyag:

Minden kisoldalba nagy szerepet kapnak a képek, hogy a felhasználók ne csak képzeletben, de ténylegesen is tudják, hogy az adott személy, csata, események hogyan is néznek ki. Amit a felhasználók láthatnak:

- térképek (pl. gyarmatosítás, szövetségi rendszerek)
- archív fotók (pl. Verdun, Lenin, bolsevik forradalom)
- grafikák és diagramok (pl. Schlieffen-terv, gorlicei áttörés).

Példa kisoldal: Az Első Világháború

Az Első Világháború kisoldal az egyetemes történelem szál egyik kulcsfontosságú része. Több év eseményeit öleli fel részletesen, képekkel, térképekkel és elemző szövegekkel.

Tartalmi felépítés:

- Előzmények: A hatalmi egyensúly megbomlása, Balkán helyzete, szövetségi rendszerek
- 1914: Szarajevói merénylet, a háború kitörése, Schlieffen-terv
- 1915: Állóháború kialakulása, harci gáz bevetése, Olaszország belépése
- 1916: Verduni és somme-i csata, Brusilov-offenzíva, Románia belépése
- 1917: Orosz forradalom, bolsevik hatalomátvétel, USA hadba lépése
- 1918: A háború vége, Compiègne-i fegyverszünet, az Osztrák-Magyar Monarchia felbomlása

A kisoldal az eseményeket időrendben követi végig, szöveges és képes magyarázatokkal kísérve, így a felhasználó könnyen elmélyülhet a korszak történéseiben.

Dizájn és vizuális elemek célja:

- vizuálisan megkülönböztesse az egyes korszakokat (egyedi háttérképek),
- a szöveges tartalom olvasható és jól tagolt legyen (sötét háttér, világos betűk, szakaszcímek),
- a képek és a szövegek harmonikusan illeszkedjenek egymáshoz

Egy darab CSS stíluslapon szerepel az összes kisoldal stílusa. Például az Első Világháború háttérképe egy korabeli háborús jelenetet ábrázol, ami hangulatilag is megerősíti az oldalon olvasottakat.

Fejlesztési lehetőségek a jövőben a kisoldalakon:

- hanganyagok, videók beillesztése,
- tesztek, feladatok beépítése tanulóknak,
- visszajelzési lehetőség (komment, kérdés, értékelés).

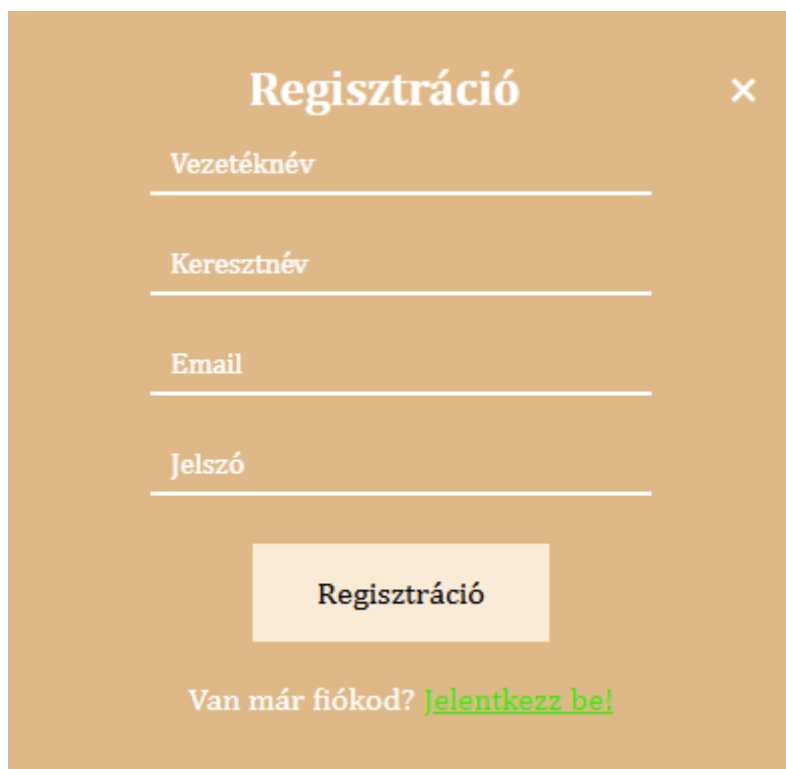
3 REGISZTRÁCIÓ ÉS BEJELENTKEZÉS

3.1 Regisztráció

A regisztrálás létfontosságú a Historia oldalban, hiszen lehetővé teszi a felhasználók számára, hogy fiókot hozzanak létre az adatok biztonságos tárolásához és hogy jegyzeteiket tudják tárolni is (6. ábra).

A regisztráció egy modális ablakban történik, amely a következő adatokat kéri:

- Vezetéknév és Keresztnév: Kötelezően kitöltendő.
- Email: Egyedi azonosítóként szolgál, kötelező.
- Jelszó: Kötelező, amelynek erősségét a rendszer ellenőrzi.

A képen egy modális ablak látható, amely a regisztrációs folyamatot szolgálja. Az ablak sárga hátterű, és a címe "Regisztráció", mellette egy "X" jelű bezáró gomb van. Az ablakban négy mező található: "Vezetéknév", "Keresztnév", "Email" és "Jelszó", mindegyik alatti egy vízszintes vonal jelöli a beviteli területet. Az mezők alatt egy "Regisztráció" feliratú gomb van. Az ablak alján a szöveg "Van már fiókod? [Jelentkezz be!](#)" látható, ahol a link zöld színnel van kiemelve.

6. ábra regisztráció

3.2 Hitelesítés

A "Regisztráció" gombra kattintva a felhasználó adatai elküldésre kerülnek, és egy emailben kapott hitelesítő kódot kell megadni a fiók aktiválásához (7. ábra). A regisztrációs folyamat két fő fájlra oszlik: egy frontend (HTML és JavaScript) és egy backend (PHP) részre.

A frontend oldalon az űrlap adatainak validálása JavaScripttel történik, például az email formátumának ellenőrzése. A "Regisztráció" gombra kattintva a regisztral.php fájlhoz kerülnek az adatok POST kéréssel.

A szerver oldalon a következő lépések történnek:

- Ellenőrzi, hogy minden kötelező mező kitöltött-e.
- Az email egyediségét a regisztralas táblában ellenőrzi.
- A jelszót a password_hash függvénnyel titkosítja.
- Egy 6 jegyű hitelesítő kódot generál, amelyet az email_hitelesites táblába ment, és emailben elküld a felhasználónak.

A felhasználó a kód megadása után aktiválhatja fiókját, amely ekkor használatra kész.



7. ábra hitelesítés

3.3 Bejelentkezés

A bejelentkezés biztosítja, hogy csak hitelesített felhasználók férjenek hozzá a rendszerhez(8. ábra).

A bejelentkezés modális ablakban érhető el, amely a következő mezőket tartalmazza:

- **Email és Jelszó:** Kötelezően kitöltendő.
- **Emlékezz rám:** Opcionális, sütik használatával működik.

8. ábra bejelentkezés

A bejelentkezés szintén két fájlra oszlik: egy HTML/JavaScript és egy PHP fájlra.

A bejelentkezés során a bejelentkezés.php fájl ellenőrzi az emailt és a jelszót (password_verify használatával).

A backend rendszere PHP alapú, MySQL adatbázissal működik, és felelős a felhasználói regisztráció, bejelentkezés, email hitelesítés, email módosítás, jelszó visszaállítás és munkamenet kezelés szerveroldali logikájáért. A rendszer a PHPMailer könyvtárat használja az email kommunikációhoz, és JSON formátumú válaszokkal kommunikál a frontenddel.

3.4 Regisztráció és bejelentkezés ablak (modal) stílusa

Az ablakok egységes és modern dizájnt követnek, amely vizuálisan is jól illeszkedik a História projekt hangulatához.

- Az ablakok div osztályt kapnak, amely középre pozicionálja az elemeket (display: flex, justify-content: center, align-items: center).
- Teljes képernyős háttér, kissé áttetsző/blur hatású belső tartalom (kontent osztály).
- A hidden osztály felel az ablak elrejtéséért – JavaScript kapcsolja be/ki.

Dizájn jellemzők:

- Háttérszín: burlywood – meleg, történelmi hangulatú árnyalat.
- Átlátszó input mezők: background: transparent, border-bottom: 2px solid white – letisztult, modern hatás.
- Fehér placeholder szöveg, ami jól látszik a sötétebb hátteren.
- A bezáró x ikon a jobb felső sarokban helyezkedik el, és hover hatásra fehérre válik
- A gombok (.gomb, .rgomb) alapból átlátszóak, hover-re kinagyulnak vagy színt váltanak – interaktív visszajelzés.

3.5 Jelszóerősség működése:

- Figyeli a jelszó mező (input#password) változását.
- Először megvizsgálja, hogy a jelszó legalább 8 karakter hosszú-e.
- Majd négy kritérium szerint pontoz:
 - Szám (0-9)
 - Kisbetű (a-z)
 - Nagybetű (A-Z)
 - Speciális karakter (!, @, # stb.)
- Ha rövidebb mint 8 karakter, akkor gyenge (piros szín)
- Ha 3 típus szerepel benne, akkor közepes (narancssárga)
- Ha mind a 4 szerepel, akkor erős (zöld szín)

A jelszóhoz tartozó mező színe is változik: borderColor piros, narancssárga vagy zöld lesz, attól függően, milyen erős (9. ábra).

```

function checkPasswordStrength(pass, msg, str) {
    pass.addEventListener("input", () => {
        if (pass.value.length > 0) {
            msg.style.display = "block";
        } else {
            msg.style.display = "none";
        }

        let pont = 0;
        let regex = /[0-9]/, /[a-z]/, /[A-Z]/, /[!@#$%^&*()_+~=-\[\]\{\};':"\|,.<>\|?]/;

        for (let i = 0; i < regex.length; i++) {
            if (regex[i].test(pass.value)) {
                pont++;
            }
        }

        if (pass.value.length < 8) {
            str.innerHTML = "gyenge";
            pass.style.borderColor = "red";
            msg.style.color = "red";
        } else {
            if (pont === 4) {
                str.innerHTML = "erős";
                pass.style.borderColor = "#26d730";
                msg.style.color = "#26d730";
            } else if (pont === 3) {
                str.innerHTML = "közepes";
                pass.style.borderColor = "orange";
                msg.style.color = "orange";
            } else {
                str.innerHTML = "gyenge";
                pass.style.borderColor = "red";
                msg.style.color = "red";
            }
        }
    });
}

```

9. ábra jelszóerősség mérő

4 ADATBÁZIS

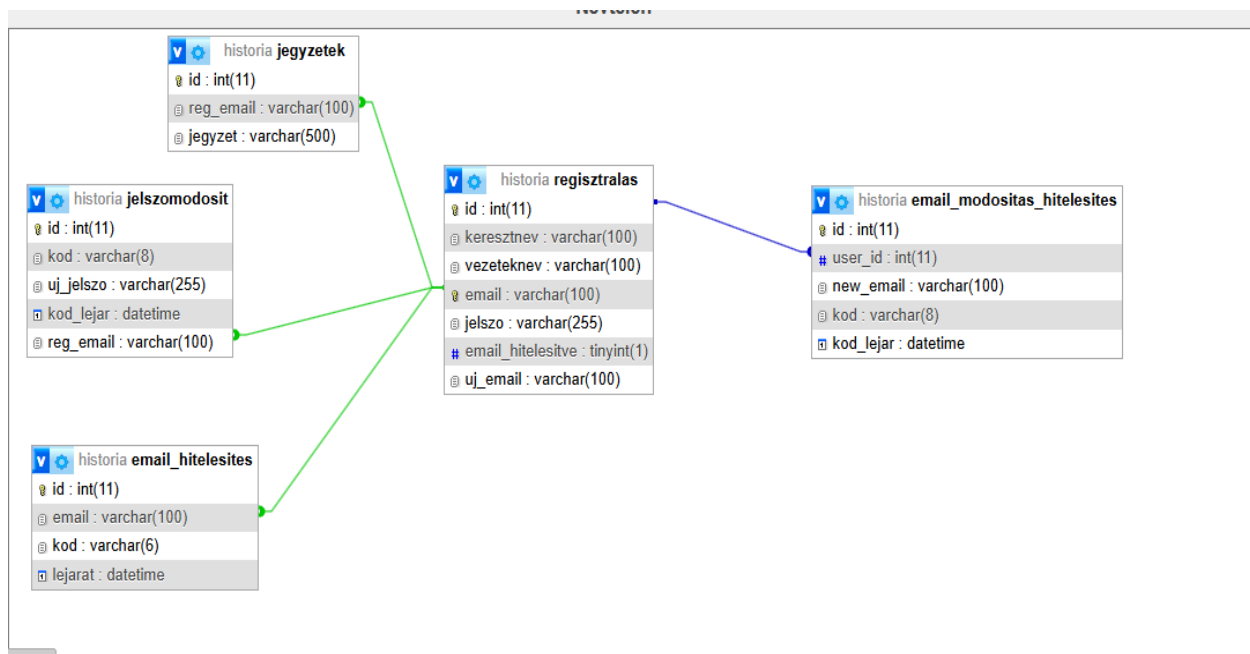
4.1 Az adatbázis bemutatása

A Historia adatbázis egy felhasználókezelési és jegyzetkezelési rendszert támogató relációs adatbázis, amely magyar nyelvű környezetre optimalizált (utf8_hungarian_ci karakterkódolással). Az adatbázis célja, hogy biztonságos, hatékony és rugalmas módon tárolja a felhasználók adatait, hitelesítési folyamatait, valamint jegyzeteiket. Az alábbiakban részletesen bemutatjuk az adatbázis szerkezetét, működését, tervezési döntéseit, valamint azok előnyeit.

4.2 Adatbázis szerkezete

Az adatbázis öt táblát tartalmaz, amelyek a következő funkciókat látják el (10. ábra):

- **regisztralas:** A felhasználók alapadatait tárolja, keresztnévet, vezetéknévet, email címet, jelszót, valamint az email hitelesítés állapotát és egy opcionális új email címet.
- **jelszomodosit:** Jelszóváltoztatási kérelmeket kezel, ideiglenes kódokkal és lejáratí időekkel.
- **email_hitelesites:** Az email címek hitelesítésére szolgáló kódokat és azok lejáratí idejét tárolja.
- **email_modositas_hitelesites:** Az email-cím módosításának hitelesítési folyamatait kezeli.
- **jegyzetek:** A felhasználók által létrehozott jegyzeteket tárolja, email cím alapján kapcsolódva a regisztralas táblához.



10. ábra adatbázis

4.3 Táblák részletes leírása

Regisztralas

Ez a tábla a felhasználók alapvető adatait tárolja. A kulcsmezők:

- id: Egyedi azonosító, automatikusan generálódik.
- keresztnev, vezeteknev: A felhasználó neve.
- email: Egyedi email cím, amely a felhasználó azonosítására szolgál.
- jelszo: Titkosított jelszó, amely biztonságos tárolást biztosít.
- email_hitelesitve: Logikai mező, amely jelzi, hogy az email cím hitelesítve van-e.
- uj_email: Opcionális mező az email cím módosításakor.

A tábla az ALTER TABLE utasításokkal bővült, hogy támogassa az email hitelesítést és az email módosítást, ami rugalmasságot biztosít a funkcionalitás későbbi bővítése során.

jelszomodosit

Ez a tábla a jelszó-visszaállítási folyamatot támogatja:

- kod: Egyedi, 8 karakteres kód a jelszó-visszaállítás azonosítására.
- uj_jelszo: Az új jelszó ideiglenes tárolására szolgál.
- kod_lejar: A kód érvényességének lejárat ideje, amely növeli a biztonságot.
- reg_email: Kapcsolat a regisztrációs táblához az email címen keresztül.

email_hitelesites

Az email hitelesítési folyamatot kezeli:

- email: A hitelesítendő email cím.
- kod: 6 karakteres hitelesítő kód.
- lejarat: A kód érvényességi ideje, amely biztosítja, hogy a kód csak korlátozott ideig használható.

email_modositas_hitelesites

Az email cím módosítását támogatja:

- user_id: Kapcsolat a regisztrációs táblához az azonosítón keresztül.
- new_email: Az új email cím.
- kod, kod_lejar: A hitelesítéshez szükséges kód és annak lejárat ideje.

jegyzetek

A felhasználók jegyzeteit tárolja:

- reg_email: Kapcsolat a regisztrációs táblához.
- jegyzet: Maximum 500 karakteres szöveg, amely a felhasználó jegyzetét tartalmazza.

4.4 Tervezése

Miért lett így tervezve?

Az adatbázis tervezése során a következő szempontokat vettük figyelembe:

Biztonság:

A jelszavak titkosítva vannak tárolva (255 karakteres mező, amely támogatja a modern hash algoritmusokat, például bcrypt).

Az email hitelesítési és jelszó visszaállítási kódok lejárat idővel rendelkeznek, így csökkentve a visszaélés lehetőségét.

A FOREIGN KEY biztosítja az adatok integritását, például egy jegyzet nem hozható létre érvénytelen email címhez.

Rugalmasság

Az ALTER TABLE utasításokkal később hozzáadott mezők (pl. email_hitelesítve, uj_email) lehetővé teszik az adatbázis funkcionalitásának bővítését anélkül, hogy a meglévő szerkezetet jelentősen módosítani kellene.

Az email módosítási folyamat külön táblában van kezelve, így a felhasználók zökkenőmentesen válhatnak e-mail-címet.

Hatékonyság:

A relációs kapcsolatok minimalizálják az adatduplikációt, és gyors hozzáférést biztosítanak a kapcsolódó adatokhoz.

Az utf8_hungarian_ci karakterkódolás biztosítja a magyar ékezetes karakterek helyes kezelését, ami különösen fontos a felhasználói adatok és jegyzetek esetében.

Felhasználóbarát működés:

A jegyzetek tábla egyszerű szerkezete lehetővé teszi a gyors és hatékony jegyzetkezelést.

Az email alapú azonosítás intuitív, mivel a legtöbb felhasználó az email címét használja bejelentkezéshez.

4.5 Előnyök

A Historia adatbázis számos előnnyel jár:

Biztonságos felhasználókezelés: A jelszó visszaállítási és email hitelesítési folyamatok modern biztonsági szabványokat követnek, védve a felhasználók adatait.

Személyre szabott élmény: A jegyzetek tábla lehetővé teszi, hogy a felhasználók saját tartalmaikat tárolják, ami növeli az alkalmazás értékét.

Skálázhatóság: A táblák szerkezete és a relációs kapcsolatok lehetővé teszik új funkciók (pl. további hitelesítési módok, új típusú tartalmak) könnyű integrálását.

Adatintegritás: A kulcsok és az egyedi mezők (pl. email) biztosítják, hogy az adatok konzisztensek és megbízhatóak maradjanak.

Magyar nyelvű támogatás: A karakterkódolás és a magyar nyelvű környezetre való optimalizálás javítja a felhasználói élményt a magyar nyelvű közönség számára.

5 BACKEND FŐ FUNCKIÓI

5.1 Adatbázis Kezelés

Az adatbázis.php fájlban definiált Adatbázis osztály felelős a MySQL kapcsolódásért. A kapcsolat a következő paraméterekkel jön létre:

Host: localhost

Felhasználó: root

Jelszó: üres (alapértelmezett fejlesztői környezetben)

Adatbázis: historia

A `set_charset("utf8")` metódus biztosítja, hogy a karakterkódolás UTF-8 legyen, így elkerülhetők a magyar ékezetekkel kapcsolatos problémák. A kapcsolat hibakezelése kivételkezeléssel (try-catch) történik: ha a kapcsolódás sikertelen, a rendszer leáll, és a hibaüzenetet naplózza (die függvény). A `getKapcsolat()` metódus biztosítja a kapcsolat lekérdezését más fájlok számára, míg a `__destruct()` metódus garantálja, hogy a kapcsolat a műveletek végeztével automatikusan lezáruljon.

5.2 Regisztráció és Email Hitelesítés

Regisztráció (regisztral.php)

A regisztráció a következő lépésekből áll:

Bemeneti Validáció:

Ellenőrzi, hogy minden kötelező mező (vezeteknev, keresztnév, email, jelszo) kitöltött-e.

Az email formátumát a `FILTER_VALIDATE_EMAIL` szűrővel validálja.

A jelszó hosszát ellenőrzi (minimum 8 karakter).

Adatbázis Ellenőrzés:

Egy `SELECT` lekérdezéssel ellenőrzi, hogy az email már szerepel-e a regisztralas táblában.

Ha igen, a rendszer hibaüzenetet küld ("Ez az email cím már regisztrálva van!").

Adatfeldolgozás:

A jelszót a password_hash függvény titkosítja a PASSWORD_DEFAULT algoritmussal.

A felhasználó adatait egy INSERT INTO parancs menti a regisztralas táblába, az email_hitelesitse mezőt FALSE értékre állítva.

Hitelesítő Kód Generálás:

Egy 6 jegyű kód generálódik (mt_rand(100000, 999999)), amely 10 percig érvényes (strtotime("+10 minutes")).

A kódot az email_hitelesites táblába menti egy INSERT INTO parancs.

Email Küldés:

Az email_kuldo.php fájl kuldo_email() függvénye küldi el a kódot a PHPMailer segítségével.

A levél tartalma szöveges formátumú, és tartalmazza a kódot valamint annak lejáratát idejét.

Munkamenet Kezelés:

Az emailt a \$_SESSION['reg_email'] változóban tárolja a további hitelesítéshez.

Email Hitelesítés (hitelesites.php és hitelesites_api.php)

A hitelesítés során a felhasználó által megadott kódot a rendszer összehasonlítja az email_hitelesites táblában tárolt kóddal:

Egy SELECT lekérdezés ellenőrzi, hogy a kód létezik-e, és nem járt-e le (kod_lejar > NOW()).

Sikeres hitelesítés esetén:

A regisztralas tábla email_hitelesitse mezőjét TRUE értékre frissíti.

Törli a kódot az email_hitelesites táblából.

Új Hitelesítő Kód Kérése (uj_kod_emailhez.php)

Ez a fájl lehetővé teszi, hogy a felhasználó új email hitelesítő kódot kérjen, ha az eredeti kód lejárt vagy elveszett:

Email Ellenőrzés:

A kérés GET paraméterként kapja meg az email címet, és ellenőrzi, hogy egyezik-e a munkamenetben tárolt \$_SESSION['reg_email'] értékkel.

Ha nem egyezik, a rendszer hibaüzenetet küld ("Érvénytelen email cím!").

Régi Kódok Törlése:

Egy DELETE parancs törli az adott emailhez tartozó összes korábbi kódot az email_hitelesites táblából, hogy elkerülje a duplikációt és biztonságosabb is.

Új Kód Generálása:

Egy új 6 jegyű kód generálódik (sprintf("%06d", mt_rand(100000, 999999))), amely 10 percig érvényes.

A kódot egy INSERT INTO parancs menti az email_hitelesites táblába.

Email Küldés:

Az email_kuldo.php fájl kuldo_email() függvénye küldi el az új kódot a felhasználónak.

Az email szöveges formátumú, és tartalmazza az új kódot valamint annak lejáratát ("A kód 10 percig érvényes.").

Sikeres küldés esetén a válasz success értéke true, és az üzenet: "Új kód elküldve!".

5.3 Bejelentkezés és Munkamenet Kezelés

Bejelentkezés (bejelentkezik.php)

Bemeneti Validáció:

Ellenőrzi, hogy az email és jelszó mezők kitöltöttek-e.

Felhasználó Ellenőrzés:

Egy SELECT lekérdezéssel lekéri a felhasználó adatait az email alapján.

Ha az email_hitelesitve mező FALSE, a rendszer hibaüzenetet küld, és a hitelesítési folyamatot kéri.

Jelszó Ellenőrzés:

A password_verify függvény összehasonlítja a megadott jelszót a tárolt hashelt jelszóval.

Munkamenet Kezelés:

Sikeres bejelentkezés esetén a felhasználó id-jét a \$_SESSION['user_id'] változóban tárolja.

Kijelentkezés (kijelentkezik.php)

A kijelentkezés törli a munkamenetet (session_destroy), és egy JSON választ küld ("Sikeresen kijelentkeztél!").

Email Módosítás és Jelszó-visszaállítás

Email Módosítás (emailmodosit.php)

Új Email Validáció:

Ellenőrzi, hogy az új email már szerepel-e a regisztralas táblában.

Az új email ideiglenesen az uj_email oszlopba kerül.

Kód Generálás:

Egy 8 karakteres kód generálódik (bin2hex(random_bytes(4))), amely 1 óráig érvényes.

A kódot a jelszomodosit táblába menti.

Email Küldés:

A kódot a PHPMailer segítségével küldi el az új e-mail címre HTML formátumban.

Kód Ellenőrzés és Frissítés:

A felhasználó által megadott kód ellenőrzése után a rendszer frissíti az e-mailt a regisztralas táblában, és törli a kódot a jelszomodosit táblából.

Jelszó-visszaállítás Kód Generálása (ujkod.php)

Ez a fájl lehetővé teszi, hogy a felhasználó jelszó-visszaállítási kódot kérjen:

Email Ellenőrzés:

A kérés POST paraméterként kapja meg az email címet, és ellenőrzi, hogy üres-e.

Egy SELECT lekérdezéssel ellenőrzi, hogy az email létezik-e a regisztralas táblában.

Ha nem létezik, hibaüzenetet küld ("Nincs ilyen email cím regisztrálva!").

Kód Generálása:

Egy 8 karakteres kód generálódik (`bin2hex(random_bytes(4))`), amely 1 óráig érvényes (`strtotime('+1 hour')`).

A kódot egy `INSERT INTO ... ON DUPLICATE KEY UPDATE` parancs menti a jelszomodosit táblába, így ha már létezik kód az adott emailhez, az frissül.

E-mail Küldés:

A PHPMailer segítségével küldi el a kódot a felhasználónak.

Az email HTML formátumú, és tartalmazza a kódot valamint annak lejáratát idejét (Érvényes: \$lejarat-ig).

A levél alternatív szöveges verziót is biztosít (AltBody), ha a felhasználó email kliense nem támogatja a HTML-t.

Sikeres küldés esetén a válasz sikeres értéke true.

5.4 Hibakezelés:

Az SQL hibák naplózásra kerülnek (`error_log`), és a felhasználó hibaüzenetet kap.

Ha az email küldés sikertelen, a PHPMailer hibáját is naplózza, és hibaüzenetet küld.

Jelszó visszaállítás (`uj_jeszo.php`)

Ez a fájl kezeli a jelszó visszaállítás véglegesítését:

Bemeneti Validáció:

Ellenőrzi, hogy a kód (`code`) és az új jelszó (`new_password`) mezők kitöltöttek-e.

Ha nem, hibaüzenetet küld ("Töltsd ki az összes mezőt!").

Kód Ellenőrzés:

Egy `SELECT` lekérdezéssel ellenőrzi, hogy a kód létezik-e a jelszomodosit táblában, és nem járt-e le (`kod_lejar > NOW()`).

Ha a kód érvénytelen vagy lejárt, hibaüzenetet küld ("Érvénytelen vagy lejárt kód!").

Új Jelszó Mentése:

Az új jelszót titkosítja (`password_hash`), és naplózza a titkosított jelszó hosszát (`error_log`).

Egy UPDATE parancs frissíti a `jelszomodosit` tábla `uj_jelszo` oszlopát, és törli a kódot (`kod = NULL, kod_lejar = NULL`).

Egy másik UPDATE parancs frissíti a `regisztralas` tábla `jelszo` oszlopát az új titkosított jelszóval, az email alapján.

Hibakezelés:

Az SQL hibák naplózásra kerülnek, és a felhasználó hibaüzenetet kap, ha bármelyik frissítés sikertelen.

Sikeres művelet esetén a válasz sikeres értéke `true`.

6 BACKEND FEJLESZTŐI DOKUMENTÁCIÓ

6.1 Fájlok és Funkciók

adatbazis.php: MySQL kapcsolat kezelése.

regisztral.php: Regisztrációs logika, email hitelesítő kódgenerálás.

hitelesites.php és hitelesites_api.php: Email hitelesítés.

uj_kod_emailhez.php: Új email hitelesítő kód generálása és küldése.

bejelentkezes.php: Bejelentkezési logika, munkamenet kezelés.

kijelentkezik.php: Kijelentkezés.

emailmodosit.php: Email módosítás.

ujkod.php: Jelszó-visszaállítási kód generálása és küldése.

uj_jeszo.php: Jelszó-visszaállítás véglegesítése.

email_kuldo.php: PHPMailer alapú email küldés.

6.2 Hibakezelés

A hibák naplózásra kerülnek (`ini_set('log_errors', 1)`), és a `php_errors.log` fájlban tárolódnak.

A kimeneti hibák ki vannak kapcsolva (`display_errors = 0`), kivéve a fejlesztési környezetben, ahol a hibák megjelenítése engedélyezett (`display_errors = 1`).

Minden SQL műveletnél `prepare` és `bind_param` metódusokat használ a rendszer az SQL injection elleni védelem érdekében.

Az email küldési hibák naplózásra kerülnek (`error_log`), és a felhasználó megfelelő hibaüzenetet kap.

6.3 Biztonság

Jelszó Titkosítás: A `password_hash` függvény biztosítja a jelszavak biztonságos tárolását.

Munkamenet Kezelés: A `session_start` és `session_destroy` függvények kezelik a munkameneteket.

Email Küldés Biztonsága: A PHPMailer SMTP hitelesítést használ (Gmail SMTP, alkalmazásjelszóval).

Adatvalidáció: A bemenetek szerveroldali validációja (pl. `FILTER_VALIDATE_EMAIL`) csökkenti a hibás adatok esélyét.

Kód Generálás Biztonsága: A kódok generálása cryptographically secure függvényekkel történik (`random_bytes`), és a lejáratási idő korlátozza az érvényességüket.

7 FEJLESZTÉS

A História weboldal fejlesztése során törekedtünk arra, hogy a tartalom és a technikai megvalósítás egységet alkosson, és a felhasználók számára egy informatív, mégis élvezetes élményt nyújtson. A munkát hárman végeztük, feladatainkat pedig igyekeztünk hatékonyan felosztani, hogy mindannyian hozzátehessük a tudásunk legjavát a projekthez.

7.1 Frontend

A felhasználói felületet HTML és CSS segítségével építettük fel, különös figyelmet fordítva az esztétikus és áttekinthető elrendezésre. Az animált idővonalat, amely a magyar és egyetemes történelem eseményeit mutatja be, szintén saját HTML és CSS megoldásokkal valósítottuk meg. Fontos volt számunkra, hogy a navigáció egyszerű, a színek és tipográfia pedig harmonikus és korhű hangulatot sugározzon. A főoldal reszponzív kialakítást kapott, hogy mobilon és asztali eszközökön egyaránt jól működjön. Későbbiekben szeretnénk az összes oldalt reszponzívra tenni.

7.2 Backend

Az oldal mögött működő backend rendszert PHP nyelven készítettük, amely lehetővé teszi a felhasználói fiókok kezelését (bejelentkezés, regisztráció).

7.3 Adatbázis

A weboldalhoz kapcsolódó adatokat MySQL adatbázisban tároljuk, amelyben többek között a felhasználói adatok, jegyzetek, és jelszó módosításhoz kapcsolódó dolgok szerepelnek. Az adatbázis szerkezete úgy lett kialakítva, hogy könnyen bővíthető legyen, amennyiben a jövőben új eseményeket vagy tartalmakat szeretnénk hozzáadni.

7.4 Fejlesztési nehézségek

A munka során számos kihívással szembesültünk, melyek hozzájárultak fejlődésünkhöz:

- Az animáció beállítása: Az idővonal vizuális megjelenítésénél kezdetben nehézséget okozott az időzítések és az elemek szinkronizálása, a szöveg konténerek idővonalra való igazítása, valamint a mobilbarát megoldás kialakítása.
- Adatbázis és backend összekötése: Az adatbázissal való kommunikáció során több hibát is tapasztaltunk a lekérdezésekben, ezek javításához mélyebben bele kellett ásunk magunkat az SQL és PHP használatába.
- Reszponzivitás és kompatibilitás: Több böngészőben és képernyőméreten is tesztelnünk kellett az oldalt, mert voltak elemek, amelyek nem úgy jelentek meg, ahogy azt szeretnénk volna, ilyen probléma volt a konténerek kilógása és ebből kifolyólag kisebb kijelző méreten való olvashatatlanság.
- Csapatmunka, összhang: A projekt kezdetén kihívás volt a feladatok szétosztása és az egységes kódkörnyezet kialakítása. Végül sikerült hatékony munkamegosztást kialakítanunk. Az összhanggal nem volt probléma, voltak időszakok mikor bátorítanunk és motiválnunk kellett egymást, de sikerült.

7.5 Tanulságok

A projekt alatt nemcsak technikai tudásunk fejlődött, hanem számos olyan készséget is elsajátítottunk, amelyek a jövőbeli projektjeink során is hasznosak lesznek:

- Csapatmunka fontossága: Háromfős csapatként dolgoztunk együtt, ami során megtapasztaltuk, mennyire lényeges a jó kommunikáció és az egymásra épülő munkavégzés. Megtanultunk hatékonyan feladatokat elosztani, egymás erősségeire építeni, és rugalmasan kezelni a felmerülő akadályokat.
- Elmélyült ismereteket szereztünk a webfejlesztés teljes folyamatáról: a tervezéstől az adatbáziskezelésen át egészen az éles felhasználói felületig.
- Hibakeresés és problémamegoldás: Több alkalommal szembesültünk váratlan hibákkal, legyen szó CSS elcsúszásokról, PHP hibaüzenetekről. Ezeket lépésről lépésre, utánajárással és kitartással oldottuk meg.
- Felhasználói élmény: Fontos tanulság volt, hogy nem elég egy oldalnak jól működnie – annak érthetőnek, esztétikusnak és könnyen használhatónak is kell lennie.

8 TESZTELÉSI DOKUMENTÁCIÓ

Tesztelés dokumentáció – „Rólunk” oldal

Teszt típusa: Funkcionális teszt

Tesztelt funkció: Rólunk oldal tartalmának, megjelenésének és elérhetőségeinek vizsgálata

Tesztelés dátuma: 2025. 04. 15.

Tesztelő neve: Hegyesi Balázs Tibor

Cél:

Annak ellenőrzése, hogy a „Rólunk” oldal tartalma megfelelően jelenik-e meg, a csapattagok adatai helyesen jelennek-e meg, és az e-mail linkek működnek-e.

Teszteset 1 – Oldal betöltése

- Lépések:
 - A főoldalról kattintsunk a navigációs sávban a „Rólunk” gombra.
- Elvárt eredmény: Az oldal betöltődik, megjelenik a „Históriásokról” címsor, alatta a bemutatkozó szöveg, valamint a három csapattag profilkérdőíve.
- Eredmény: Sikeres
- Megjegyzés: Az oldal tartalma teljesen betöltődött, a stílus és az elrendezés megfelelő.

Teszteset 2 – Csapattagok megjelenítése

- Lépések:
 - Görgessünk le az oldal aljára.
 - Ellenőrizzük, hogy mindhárom csapattag neve, képe és elérhetősége szerepel-e.
- Elvárt eredmény: A csapattagok neve, titulusuk és elérhetőségük helyesen jelenik meg, a képek jól láthatók.
- Eredmény: Sikeres
- Megjegyzés: A képek és adatok megfelelően jelentek meg.

Teszteset 3 – Elérhetőségi linkek működése

- Lépések:
 - Kattintsunk rá egy csapattag e-mail címére.
 - Elvárt eredmény: Az alapértelmezett levelezőprogram megnyílik az adott e-mail címmel a címzett mezőben.
- Eredmény: Sikeres
- Megjegyzés: Minden e-mail link működőképes volt (mailto: formátum jól működik).

Összegzés:

A „Rólunk” oldal minden vizsgált funkciója hibamentesen működött. A tartalom informatív és esztétikus, az elérhetőségek aktívak, és könnyen elérhetők bármilyen felhasználói kérdés vagy észrevétel esetén.

Ez a teszt csak egy funkcionális teszt volt, amikor készítettük a weboldalt rengeteget teszteltünk, amennyiben hibát találtunk, igyekeztük kijavítani azt.

9 RENDSZERKÖVETELMÉNYEK

9.1 Minimum rendszerkövetelmények

- Operációs rendszer: Windows 7 / macOS 10.11 / Linux disztribúciók (Ubuntu 18.04+)
- Böngésző:
 - Google Chrome 85+
 - Mozilla Firefox 80+
 - Microsoft Edge 85+
- Internetkapcsolat: stabil, legalább 5 Mbps sebességű kapcsolat
- Képernyőfelbontás: minimum 1280×720 px (HD)
- Hardver:
 - Processzor: Dual-Core 1.5 GHz
 - RAM: 2 GB
 - Tárhely: 100 MB szabad hely (cache + jegyzetek lokális tárolásához)

9.2 Ajánlott rendszerkövetelmények

- Operációs rendszer: Windows 10 vagy 11 / macOS 12+ / Frissített Linux disztró
- Böngésző:
 - Google Chrome (legfrissebb verzió)
 - Firefox (legfrissebb verzió)
 - Safari (14+)
- Internetkapcsolat: legalább 15 Mbps (az interaktív elemek gyors betöltése érdekében)
- Képernyőfelbontás: 1920×1080 px (Full HD) vagy magasabb
- Hardver:
 - Processzor: Quad-Core 2 GHz vagy erősebb
 - RAM: 4 GB+
 - Tárhely: 200 MB szabad hely

FORRÁSOK ÉS HASZNÁLT PROGRAMOK

Források

- Pinterest: - <https://hu.pinterest.com/>
- Zanza TV: - <https://zanza.tv/>
- Múlt-kor: - <https://mult-kor.hu/>
- Rubicon: - <https://rubicon.hu/hu>
- Történelem füzetek – tanórai jegyzetek
- Atlasz – Történelmi atlasz középiskolásoknak
- YouTube - <https://www.youtube.com/>

Használt programok

- Paint
- GIMP
- Microsoft Word
- Jegyzettömb
- Visual Studio Code
- GitHub
- Git Extensions
- XAMPP
- HTML
- CSS
- JavaScript
- PHP
- Bootstrap
- Discord