

Weiss Manfréd Szakgimnázium, Szakközépiskola és Kollégium

Szakdolgozat

Bookli

Témavezető:

Kovács László

Készítette:

Kálmán László

Mészáros Tibor
Márton

Budapest, 2025

Tartalomjegyzék

Bevezetés:

Témaválasztás.....	3
Miért pont „Bookli”?	3
Célkitűzés.....	3
Kinek szánjuk a weboldalt.....	3

Fejlesztői dokumentáció:

Fejlesztői eszközök:.....	4
Fejlesztői program nyelvek: :.....	4

Bevezetés

Témaválasztás:

A csapat tagjai körében gyakran felmerülő probléma, hogy az általuk elolvasott könyvekkel kapcsolatos véleményeiket, az olvasás során elért oldalszámot, vagy egyszerűen a könyvek megosztását barátaikkal nem tudják megfelelő platformon megvalósítani és így született meg a Bookli ötlete.

Miért pont „Bookli”?

Több név is felmerült lehetőségként, például: Bookit, Bookin, ReadWithMe és BookFriends, hogy csak a kiemelkedőbbeket említsük. A csapat azonban közös megegyezéssel végül a "Bookli" elnevezés mellett döntött.

Célkitűzés:

Célunk egy felhasználóbarát könyves weboldal létrehozása, amely lehetővé teszi a felhasználók számára a könyvek keresését, értékelését és véleményezését. A regisztrációval járó alapvető funkciók mellett további előnyöket is kínálunk, például egyedi felhasználói profil létrehozásának lehetőségét. Ezáltal a felhasználók beléphetnek saját profiljukba, ahol könnyedén hozzáférhetnek a könyveikhez és korábbi véleményezéseikhez. A regisztrált tagok számára lehetőség nyílik a könyvek értékelésére, tapasztalataik megosztására és ajánlások megtételére, melynek köszönhetően egy aktív és interaktív közösség alakulhat ki az oldalon.

Kinek szánjuk a weboldalt:

Weboldalunkat minden könyvrajongó számára ajánljuk, akik szeretnék bővíteni ismereteiket, vagy egyszerűen csak élvezni kívánják az irodalom világát.

Fejlesztői dokumentáció

Fejlesztői eszközök:

Fejlesztőeszközök használata könyves weboldalunk elkészítéséhez

A könyves weboldalunk fejlesztése során több különböző, hatékony fejlesztőeszközt alkalmazunk, amelyek megkönnyítik a weboldal kialakítását, az adatok kezelését és a tesztelést.



Visual Studio Code:

A Visual Studio Code egy könnyen használható, ingyenes és nyílt forráskódú kódszerkesztő, amelyet a Microsoft fejlesztett ki. Ez az eszköz számos hasznos funkcióval rendelkezik, például szintaxiskiemeléssel, intelligens kódkiegészítéssel, hibakereséssel és kódrészletek kezelésével, amelyek mind hozzájárulnak a hatékony fejlesztéshez.



XAMPP:

A XAMPP egy integrált fejlesztői környezet, amely tartalmazza az Apache webszervert, a MySQL adatbázis-kezelőt és a PHP-t. Ez lehetővé teszi számunkra, hogy a weboldalt lokálisan fejlesszük és teszteljük a saját számítógépünkön, anélkül hogy valós webszerverre lenne szükség. Ez különösen hasznos a fejlesztési folyamat során, mivel gyors és biztonságos környezetet biztosít.



phpMyAdmin:

A phpMyAdmin egy webalapú eszköz, amely lehetővé teszi az adatbázisok egyszerű kezelését és karbantartását. Segítségével könnyedén végezhetünk műveleteket, például adatbázisok létrehozását, táblák szerkesztését vagy adatok importálását és exportálását.



MySQL:

A MySQL egy hatékony és megbízható adatbázis-kezelő rendszer, amelyet a weboldalunkon tárolt adatok, például könyvek, felhasználói információk és egyéb kapcsolódó adatok kezelésére használunk. Ez az eszköz biztosítja az adatok gyors és biztonságos elérését.

Miért választottuk ezeket az eszközöket?

Ezeket az eszközöket azért választottuk, mert már korábban is dolgoztunk velük, így jól ismerjük a működésüket. Emellett széles körben elérhetők, ingyenesen használhatók, és kiváló támogatást nyújtanak a webfejlesztés különböző szakaszaiban. Segítségükkel hatékonyan fejleszthetjük és tesztelhetjük weboldalunkat, valamint egyszerűen kezelhetjük az adatbázisokat.

Fejlesztői program nyelvek:



HTML (HyperText Markup Language):

A HTML a weboldalak alapvető építőköve, amely a tartalom szerkezetét határozza meg. Segítségével definiálhatók a weboldal elemei, például szövegek, képek, linkek, táblázatok és űrlapok. A HTML biztosítja a weboldal alapvető vázát, amelyet más technológiák, például CSS és JavaScript egészítenek ki.



CSS (Cascading Style Sheets):

A CSS a weboldalak megjelenésének és stílusának meghatározására szolgál. Segítségével testreszabhatók az elemek színei, betűtípusai, elrendezései és egyéb vizuális tulajdonságai. A CSS lehetővé teszi a weboldalak esztétikus és egységes megjelenésének kialakítását, miközben elkülöníti a tartalmat a megjelenéstől.



JavaScript:

A JavaScript egy programozási nyelv, amely dinamikus és interaktív funkciókat ad a weboldalakhoz. Például lehetővé teszi az űrlapok validálását, animációk létrehozását, vagy a felhasználói interakciók kezelését. A JavaScript a böngészőben fut, és közvetlenül a HTML és CSS elemekkel dolgozik, hogy a weboldalak élményszerűbbé váljanak.



PHP (Hypertext Preprocessor):

A PHP egy szerveroldali programozási nyelv, amely dinamikus weboldalak létrehozására szolgál. A PHP segítségével adatokat dolgozhatunk fel, adatbázisokkal kommunikálhatunk, és dinamikus generálhatunk HTML-tartalmat. Mivel a PHP a szerveren fut, a felhasználó csak az eredmény kapott HTML-t látja a böngészőjében.

Hogyan működnek együtt?

Ezek a technológiák együtt alkotják a modern weboldalak alapját. A HTML biztosítja a tartalom szerkezetét, a CSS gondoskodik a megjelenésről, a JavaScript dinamikus funkciókat ad hozzá, míg a PHP a szerveroldali logikát és adatkezelést végzi. Együttműködésük révén teljes értékű, interaktív és vizuálisan vonzó weboldalak hozhatók létre.

Fejlesztői Hardware:

Processzor: Intel(R) Core(TM) i3 CPU 550 @ 3.20GHz 3.20 GHz

Memória mérete: 8,00 GB

Rendszer típusa: 64 bites Windows 10 operációs rendszer, x64-alapú processzor

Videókártya neve: NVIDIA GeForce GTX 650 Ti

Hibakezelési esetek

1. Regisztrációs folyamat tesztelése (reg_ir.php)

1.1 Hiányzó felhasználónév:

```
if($_POST['username']=="" )  
    die("<script> alert('Nem adtad meg a felhasználóneved!') </script>");
```

Leírás: Ha a username mező üresen marad. **Elvárt eredmény:** Egy alert ablak jelenik meg a "Nem adtad meg a felhasználóneved!" üzenettel.

1.2 Hiányzó keresztnév:

```
if($_POST['keresztnev']=="" )  
    die("<script> alert('Nem adtad meg a keresztned!') </script>");
```

Leírás: Ha a keresztnév mező üres. **Elvárt eredmény:** Egy alert jelenik meg: "Nem adtad meg a keresztned!".

1.3 Hiányzó vezeteknév:

```
if($_POST['vezeteknev']=="" )  
    die("<script> alert('Nem adtad meg a vezetékned!') </script>");
```

Leírás: Ha a vezeteknev mező nincs kitöltve. **Elvárt eredmény:** Egy alert jelenik meg: "Nem adtad meg a vezetékned!".

1.4 Hiányzó email cím:

```
if($_POST['email']=="" )  
    die("<script> alert('Nem adtad meg az email címed!') </script>");
```

Leírás: Ha a email mező üres. **Elvárt eredmény:** Egy alert jelenik meg: "Nem adtad meg az email címed!".

1.5 Hiányzó jelszó:

```
if($_POST['password']=="" )  
    die("<script> alert('Nem adtad meg a jelszavad!') </script>");
```

Leírás: Ha a password mező nincs kitöltve. **Elvárt eredmény:** Egy alert jelenik meg: "Nem adtad meg a jelszavad!".

1.6 Sikeres regisztráció:

Leírás: Minden mező megfelelően ki van töltve. **Elvárt eredmény:** Az adatok bekerülnek az adatbázisba, majd a felhasználót a bejelentkezési oldalra irányítja a rendszer

2. Bejelentkezési folyamat tesztelése (login_ir.php)

2.1 Hibás bejelentkezési adatok:

```
session_start() ;
include( "kapcsolat.php" ) ;
$userek = mysqli_query( $adb , "
    SELECT * FROM user
    WHERE (uemail='$_POST[email]' OR username='$_POST[email]')
    AND password = '$_POST[pw]'
" ) ;

if( mysqli_num_rows($userek)==0 )
{
    print "<script> alert('Hibás belépési adatok!') </script>" ;
    print "<script> parent.location.href='./?p=login' </script>";
}
```

Leírás: Ha a megadott email vagy felhasználónév nem szerepel az adatbázisban, vagy a jelszó hibás.

Elvárt eredmény: Egy alert jelenik meg: "Hibás belépési adatok!", majd a bejelentkezési oldalra irányítja a felhasználót.

2.2 Sikeres bejelentkezés:

```
else
{
    $user = mysqli_fetch_array($userek) ;
    $_SESSION['uid'] = $user['uid'] ;
    $_SESSION['username'] = $user['username'] ;
    $logip = $_SERVER['REMOTE_ADDR'];
    $logsess = substr(session_id(), 0 , 8);

    mysqli_query($adb,
        "INSERT INTO login (    logid,    logdate,    logip,    logsession,        luid)
        VALUES                (    NULL,        NOW(), '$logip',    '$logsess',    '$_SESSION[uid]')");

    mysqli_close( $adb ) ;

    print "<script> parent.location.href = './' </script>" ;
}
```

Leírás: Ha a felhasználónév/email és jelszó helyes. **Elvárt eredmény:** A rendszer belépteti a felhasználót, bejegyzí a login történetet az adatbázisba, majd a főoldalra irányítja.