

BKSZC Pogány Frigyes Technikum

# Tanfolyamszervezés

szoftverfejlesztő és -tesztelő vizsgaremek

2025. május

Készítette:

Bulyáki Dóra

Jambrik Viola

Barth Maja

# A feladat leírása

Ez a dokumentáció a **Cattalodge** nevű webalkalmazás bemutatására szolgál. A projekt célja, hogy egy felhasználóbarát online platformot hozzon létre, ahol a felhasználók információkat találhatnak a cicákról, regisztrálhatnak, bejelentkezhetnek és kapcsolatba léphetnek a szolgáltatóval. Az alkalmazás összekapcsolja az Express.js backend-et egy MongoDB adatbázissal, valamint HTML, CSS és JavaScript technológiákkal készült frontenddel.

## Specifikáció

Ez a nyaraló hely még építés alatt van, a tervünk az, hogy egy nyaraló legyen, ahol egy szoba van kiadva egy hétre macskákkal.

Elsőként a főoldalra dob fel minket, ahol a nyaralóról a következő információkat lehet böngészni:

Rólunk

Macskáink

A szobáról képek

A szobáról egyéb információk

Majd a főoldalon lehet regisztrálni/bejelentkezni, A fiókról a következő adatokat kell tárolni:

A fiók neve

A fiók jelszava

A fiók profilképe (a fiók készítésnél lehet kiválasztani az adott képekből)

A fiók e-mail címe

A regisztráció/bejelentkezés után feloldódik a foglalás lehetőség. A szobáról való információs mező fölött megjelenik egy “foglalj most” gomb, amire, ha rákattintasz, legörget téged a foglalás részleghez. A foglalás részlegen van egy naptár, ahol ki lehet választani az adott hetet, amikor a felhasználó szeretné a szobánkat kivenni. Ez az adat később tárolva lesz. (A már lefoglalt heteket nem lehet kétszer kivenni).

## **Használt** Technológiák **és Eszközök**

A projekt során a következő technológiákat és eszközöket használtuk:

* Backend**:** Express.js
* Adatbázis**:** Mysql
* Frontend**:** HTML, CSS, JavaScript
* Fejlesztőikörnyezet**:** Visual Studio Code

Egyébeszközök**:** Git, Bootstrap

## Feladatkörök és Csapattagok

A csapattagok közötti feladatmegosztás a következőképpen alakult:

* Bulyáki Dóra: HTML fejlesztés, Backend fejlesztés és adatbázis kezelés
* Jambrik Viola**:** HTML fejlesztés, CSS fejlesztés, frontend design.
* Barth Maja**:** HTML és JavaScript funkciók implementálása, de főképpen a PHP-ban segített.

## GitHub repository

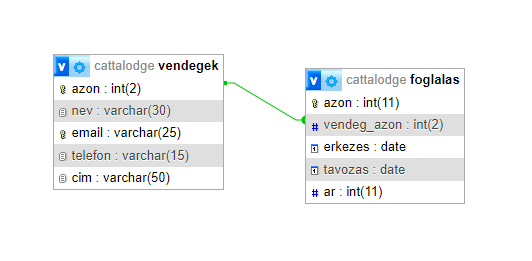
Az alkalmazások kódjai ebben a repository-ban érhetők el. Elég a bennelévő .zip filet letölteni, minden más csak azért lett megtartva, hogy a folyamatos munkának legyen nyoma:

<https://github.com/cattalodge/cattalodge>

# Bulyáki Dóra: Dokumentáció

## Az adatbázis elkészítése

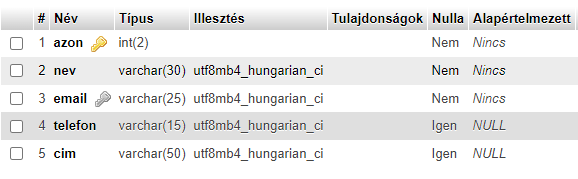
Az adatbázist a MySQL adatbázis-kezelővel és a PHPMyAdmin programmal készítettük el. Az adatokat kettő táblára bontottuk:



## A vendégek adatai

Minden vendégnek kell egy azonosító (azon), amely az elsődleges kulcs lesz. Ezen kívül szükség van a vendég nevére (nev), az email címére(email), telefonszámára(telefon) és a vendég címére (cím).

Ennek megfelelően a vendegek tábla szerkezete:



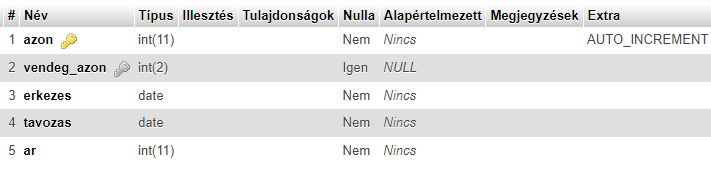
Az alkalmazás teszteléséhez az alábbi tesztadatokat vittük be:



## A foglalás adatai

Minden vendégnek kell egy azonosító (azon), amely az elsődleges kulcs lesz. Ezen kívül szükség van a vendég azonositójára(vendeg\_azon). A foglalás kezdete nem lehet üres (erkezes), a távozás sem(tavozas), és az apartman ára sem (ar).

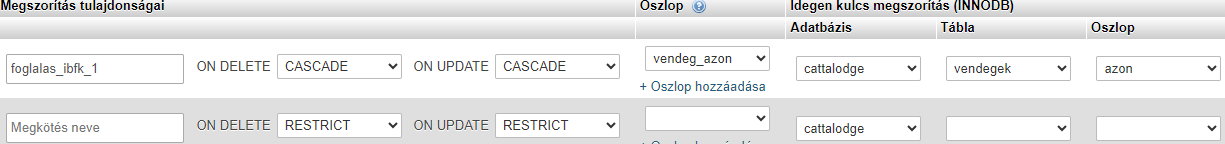
Ennek megfelelően a vendegek tábla szerkezete:



Az alkalmazás teszteléséhez az alábbi tesztadatokat vittük be

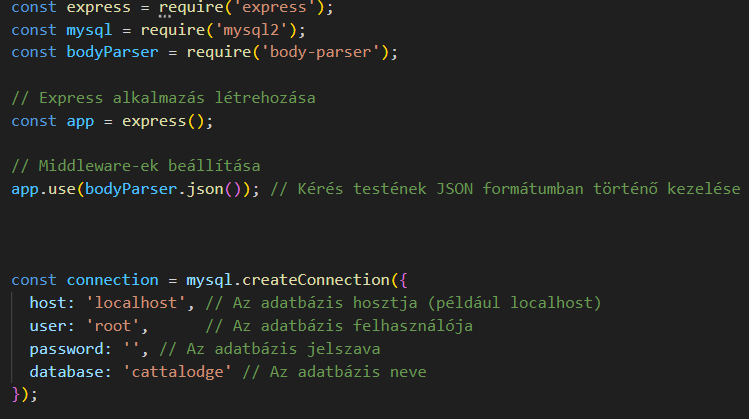


A két tábla közötti kapcsolatot prezentálnunk kell külső kulcs megadásával az altábla részéről, mellyel hozzáfűzzük a főtáblához.



## **Backend** Fejlesztés **(Express.js és Mysql)**

A projekt backend része Express.js alapú, amely összekapcsolódik egy Mysql adatbázissal. A Mysql-ben tároljuk a felhasználói adatokat, például a neveket és az e-mail címeket. Az adatbázis csatlakoztatásához és kezeléséhez az alábbi kódot használtuk:



Ez a kód egy egyszerű Express alkalmazás, amely Mysql adatbázist használ felhasználók tárolására. Három fő funkciót valósít meg:

1. Felhasználókhozzáadása: POST kérés segítségével új felhasználókat lehet hozzáadni.
2. Felhasználóklekérése: GET kérés segítségével lekérhetők az összes felhasználó adatai.
3. Alapértelmezettoldal: GET kérés a root URL-en, amely egy egyszerű üzenetet ad vissza.

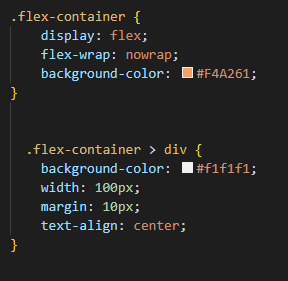
Az alkalmazás Mysql adatbázis használatával működik.

## **Frontend Fejlesztés (HTML és CSS)**

A frontend oldal HTML és CSS használatával készült. Az alábbiakban bemutatjuk a "Rólunk" oldal HTML és CSS kódját, amely a csapattagokat mutatja be, valamint a fő navigációs menüt és egyéb oldalakat tartalmaz.



Csapattagok Flexbox elrendezése  
A .flex-container használatával a három csapattag képe és bemutatkozása egy sorba rendeződnek, és a gap: 20px távolságot biztosít köztük. Minden csapattag egy div elembe van helyezve, amely tartalmaz egy képet és egy leírást a csapattagról.



Ez a CSS kódrészlet a .flex-container és annak gyermek elemeinek (a benne lévő div elemek) stílusait definiálja, és a Flexbox elrendezés használatával alakítja a weboldal dizájnját.

Ez a CSS kód egy rugalmas elrendezést biztosít a weboldalon, ahol a gyerek elemek (a div elemek) egymás mellett, sorban helyezkednek el a szülő .flex-container elemben. A Flexbox segítségével a weboldal elemei könnyen rendezhetők, és a tartalom reszponzív módon, dinamikusan alkalmazkodik az ablak méretéhez.

A gyermek elemek szélessége fixált, 100 pixel, és mindegyik között 10 pixelnyi távolság van. Az elemek tartalmát középre igazítja a text-align: center stílus, így ha bármilyen szöveg, kép vagy egyéb inline elem kerül bele, az középre lesz igazítva. A világos szürke háttérszín és a narancssárga szülő háttérszín vizuálisan jól elkülöníti az elemeket, így áttekinthető és rendezett elrendezést biztosít.

Ha a flex-wrap: nowrap; helyett wrap lenne beállítva, akkor a gyermek elemek további sorokba törhetnének, ha nem férnek el egy sorban, de így mindegyik elem egy sorban marad.

Ez a projekt lehetőséget biztosított arra, hogy fejlesszük a [pl. csapatmunka, problémamegoldás, fejlesztési készségek] terén szerzett tapasztalatainkat. Összességében egy jól megtervezett és sikeres projektet tudtunk létrehozni.

# Jambrik Viola: Dokumentáció

## HTML részek

### Main.html:

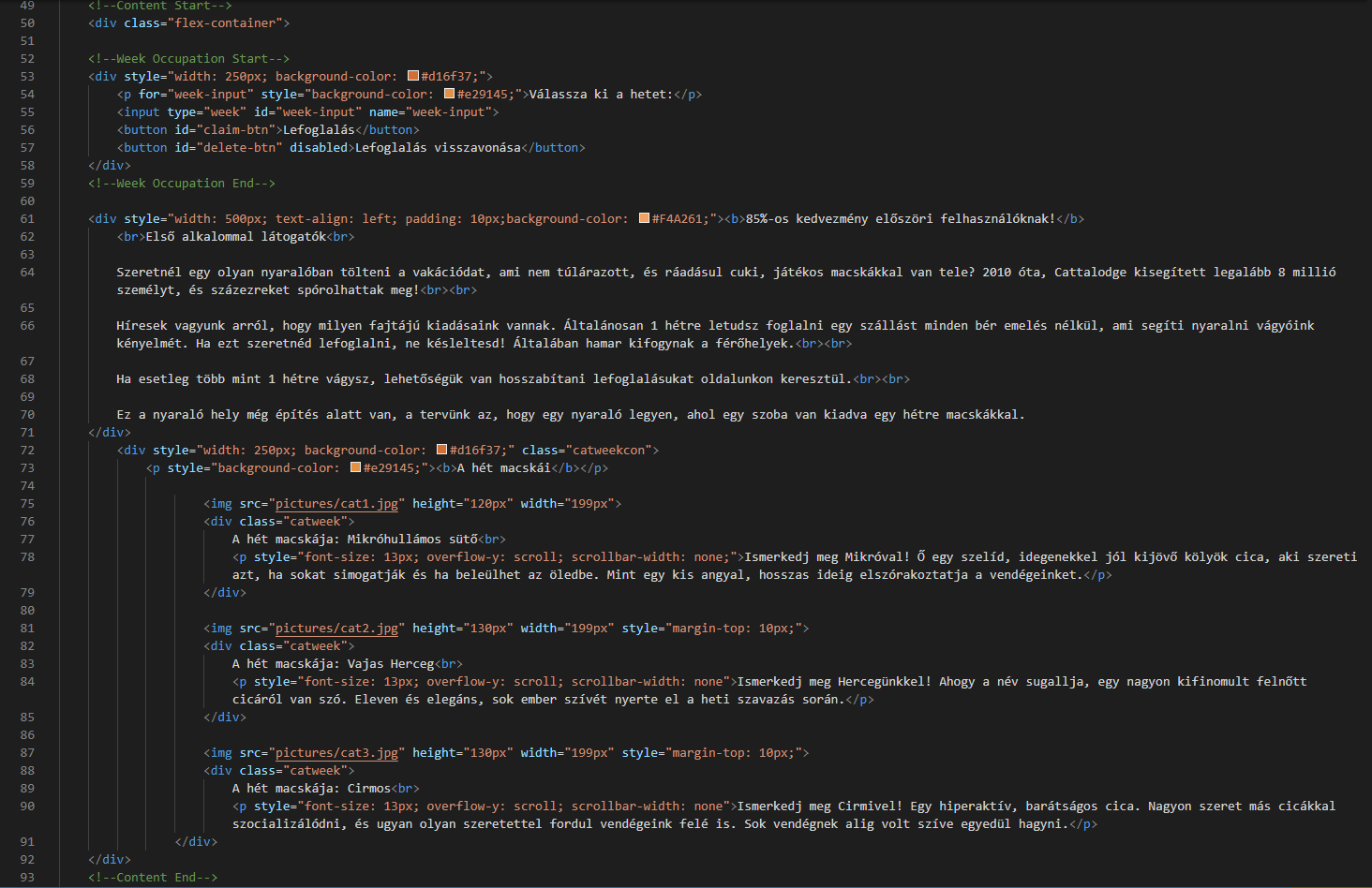
Az oldal fejlécében található a "Cattalodge" logója, amit a <img> elem tölt be. A logó mellett a fő cím és egy alárendelt címsor szerepel, amelyek bemutatják az oldal lényegét. A navigációs sáv középre igazított linkeket tartalmaz, amelyek különböző oldalakra vezetnek, mint például a főoldal, a bejelentkezés, a regisztráció, a cicák bemutatása és a "Rólunk" oldal, ami összefoglalja, hogy ki mit csinált a projekt során.



 A tartalom egyik részében lehetőséget nyújtunk a látogatóknak, hogy kiválasszák, melyik hetet szeretnék lefoglalni nyaralásukra, egy dátumválasztó mező segítségével, amely gyorsítja a folyamatot. Mellette egy figyelemfelkeltő promóciós szöveg jelenik meg, amely első alkalommal látogatóknak szóló kedvezményt kínál. A szöveg arra hívja fel a figyelmet, hogy az oldal által biztosított szállás megfizethető, és a látogatók élvezhetik a barátságos és játékos macskák társaságát.

Az oldal tartalmaz egy további részt, amely bemutatja a "hét macskáit". Minden macskához tartozik egy kép és egy leírás, amely ismerteti a cicáinkat. Ezek a tömbök tömörítettek, lehet őket görgetni.

A weboldal láblécében szerepelnek az elérhetőségi információk és a szerzői jogi nyilatkozat. Az itt feltüntetett telefonszámon keresztül a látogatók kapcsolatba léphetnek a támogatási csapattal, ha problémájuk adódna.



### Login.html:

Ez a weboldal a bejelentkezési funkció megvalósítását mutatja be, amely lehetővé teszi a felhasználók számára, hogy személyes fiókjukba lépjenek és ott foglaljanak. A designért felelős stílusokat két külső CSS-fájl határozza meg: a user.css és a responsive.css.

Az oldal középpontjában egy konténer helyezkedik el, amely magába foglalja a bejelentkezési űrlapot. Az űrlap egyszerű, felhasználóbarát kialakítással rendelkezik, amely biztosítja, hogy a felhasználók könnyen megértsék és használják. Két fő mezőt tartalmaz: egy felhasználónév beviteli mezőt, amely szöveget vár, valamint egy jelszó mezőt, amely titkosított adatot fogad. Mindkét mező kötelezően kitöltendő, és segít a bejelentkezési folyamat biztonságos működésében.

A folyamat végén található egy beküldési gomb, amely az adatokat POST metódussal küldi el feldolgozás céljából a "main.html" fájlhoz. Az űrlap ezen kívül tartalmaz két linket is, amelyek további navigációs lehetőségeket nyújtanak. Az egyik a regisztrációs oldalra vezeti a látogatókat, ha még nincs fiókjuk, míg a másik a főoldalra irányítja vissza őket.



### Register.html:

Ez a weboldal a regisztrációs funkcióját jeleníti meg, amely lehetővé teszi a felhasználók számára, hogy létrehozzák saját fiókjukat, és hozzáférést nyerjenek az oldal szolgáltatásaihoz. Az oldal címe, "Cattalodge • Regisztráció", egyértelműen utal a weboldal funkciójára, és a reszponzív megjelenés biztosítása érdekében meta címkékkel van ellátva.

Az oldal fő eleme egy központi konténer, amely tartalmazza a regisztrációs űrlapot. Ez az űrlap egyértelműen strukturált és felhasználóbarát kialakítással rendelkezik. Az első mezőben az e-mail címet kell megadni, amit a rendszer az azonosításhoz használ. Ezt követi a felhasználónév mező, amely lehetővé teszi a felhasználók számára, hogy személyre szabott nevet válasszanak.

A jelszó megadása két lépésben történik, hogy garantálja a pontos adatbevitel és a biztonságos bejelentkezés feltételeit. Először a jelszót kell beírni, majd ugyanazt megismételni a következő mezőben.

Az űrlap végén található egy "Regisztrálás" gomb, amely beküldi a felhasználói adatokat POST metódussal a "login.html" fájlba, ahol feldolgozásra kerülnek. Az oldal további navigációs lehetőségeket is kínál. Egy link segítségével a felhasználók a már meglévő fiókjukba léphetnek be a bejelentkezési oldalon keresztül, vagy visszatérhetnek a főoldalra.



## CSS részek

### style.css:

Ez a CSS fájl a főoldalak stílusát határozza meg, kivéve a bejelentkezési és regisztrációs oldalakat. Az html és body elemek beállításai egy háttérképet és színeket hoznak létre. A header elem lineáris színátmenetet használ, amely modern megjelenést biztosít az oldal fejlécének. A logó méretezését a logo osztály vezérli, amely 200 pixel szélességet állít be. Ezen felül a title és minititle osztályok felelősek a címsorok pozicionálásáért és formázásáért. Az abszolút pozicionálás megfelelő elrendezést tesz lehetővé.

A dropdown osztály jobb oldalra igazított, lebegő pozíciót kínál, amely egyszerű navigációs funkciót hoz létre. A flex-container és flex-container2 osztályok a flexbox elrendezést használják. Ezek az osztályok különböző szekciókat hoznak létre az oldalon. Az occupied osztály piros színnel és fehér szöveggel kiemeli a foglalt szálláshelyeket. A button osztály stílusai egyszerűen meghatározzák a gombok elrendezését és margóját.

A body-section-container osztály a láblécben alkalmazott felépítést hozza létre, középpontba helyezve a tartalmat és vertikálisan elrendezve a különböző elemeket. A catweek és catdesc osztályok az "A hét macskái" részletezéséhez és megjelenítéséhez szolgálnak, amelyeket szintén flexbox elrendezés tart rendben. A catweekcon és a catdesc osztályok görgethető tartalommal rendelkeznek, ugyanakkor a görgetésre szóló mezők rejtve vannak, hogy tisztább vizuális élményt nyújtsanak. A borderdesc osztály fehér szegéllyel hangsúlyozza ki a tartalom bizonyos elemeit.

### responsive.css:

Biztosítja, hogy az oldal rendesen jelenjen meg bármilyen ablak méreten.

### user.css:

Az oldal teljes hátterének stílusát a body elem definiálja. A magasság 100%-os beállítása biztosítja, hogy az oldal teljes képernyőt lefedjen, míg az overflow: hidden kiküszöböli a görgetési sávokat. A.center osztály kulcsszerepet játszik, mivel ezzel helyezi középre az űrlapot mind függőlegesen, mind vízszintesen. A fejléc, amelyet a center h1 osztály tesz lehetővé, középre igazítja a címet, és egy alsó fekete keret segítségével hangsúlyozza.

A mezők beviteli területei letisztultak, és a focus effektus révén finom vizuális visszajelzést biztosítanak, amikor a felhasználó interakcióba lép velük. A beküldési gomb stílusait az input[type="Submit"] osztály határozza meg. A gombok nagy mérete és kiemelkedő színezése biztosítja a könnyű észlelhetőséget. A navigációs linkek, amelyeket a .signup\_link osztály és annak gyermek elemeik formáznak. Az inaktív állapot szintén letisztult, míg a hover állapot hangsúlyosabbá válik a szöveg aláhúzásával. A .HomeAbout osztály külön szekciókat biztosít, amelyek arányosan illeszkednek az oldal méreteihez.

## JavaScript rész

### script.js:

A script egy DOMContentLoaded eseménnyel kezdődik, amely biztosítja, hogy a kód csak akkor fusson le, miután az oldal teljesen betöltődött. Az első funkció a legördülő menü működtetéséhez kapcsolódik. A dropdownTrigger változó a menü aktiválásáért felelős elemet azonosítja, míg a dropdownMenu maga a menü elemet tárolja. Egy "click" eseményfigyelő segítségével ez a kód megváltoztatja a legördülő menü megjelenítésének állapotát (blokkolt vagy rejtett). Emellett a window elemen egy másik eseményfigyelő gondoskodik arról, hogy a felhasználó kattintásával kívül eső területeken automatikusan bezáródjon a menü.

A hétfoglalási funkció egy Set adattípusra épül, amely az elfoglalt hetek azonosítását szolgálja. A weekInput változó azonosítja azt az űrlapmezőt, amelyben a felhasználó kiválaszthatja a hét időpontját, míg a claimButton és deleteButton gombok a foglalási és törlési műveletekhez kapcsolódnak. A weekInput mező egy "change" eseményfigyelőt tartalmaz, amely ellenőrzi, hogy az aktuálisan kiválasztott hét már foglalt-e. Ha igen, a mező egy occupied osztályt kap, amely vizuálisan kiemeli, hogy az adott hét nem elérhető. Emellett ez a művelet engedélyezi a törlés gombot, míg az ellenkező esetben letiltja azt.

A "foglalás" gombra kattintva a claimButton eseményfigyelő működik. Ha a felhasználó egy még szabad hetet választ ki, az hozzáadódik az occupiedWeeks halmazhoz, és egy figyelmeztető üzenet tájékoztatja a sikeres foglalásról. Ha azonban a hét már foglalt vagy nincs kiválasztva, a felhasználó hibaüzenetet kap. A törlés gombhoz tartozó deleteButton eseményfigyelő ellenőrzi, hogy a kiválasztott hét foglalt-e, majd eltávolítja azt az occupiedWeeks halmazból. Ha a hét sikeresen törlődik, a mezőhöz kapcsolódó vizuális kiemelés megszűnik, és a törlési gomb ismét letiltódik.

Készítette:  
Bulyáki Dóra

Jambrik Viola

Barth Maja