BMSZC Petrik Lajos Két Tanítási Nyelvű Vegyipari, Környezetvédelmi és Informatikai Technikum

1146 Budapest, Thököly út 48-54.

**VIZSGAREMEK**

***Vendégház Vadaspark***

Készítette: Szebeni Zita és Magyarosi Richárd

Tartalom

[2 Bevezetés 1](#_Toc197278217)

[2.1 Feladat leírás 1](#_Toc197278218)

[2.2 A projekt két komponensből épül fel 1](#_Toc197278219)

[2.2.1 Asztali alkalmazás 1](#_Toc197278220)

[2.2.2 Webes alkalmazás 2](#_Toc197278221)

[2.3 Miért ezt a projekt témát választottam? 2](#_Toc197278222)

[2.4 A felhasznált ismeretek 3](#_Toc197278223)

[2.5 A felhasznált szoftverek 4](#_Toc197278224)

[3 Felhasználói dokumentáció 4](#_Toc197278225)

[3.1 A program általános specifikációja 4](#_Toc197278226)

[3.2 Szoftver követelmények 5](#_Toc197278227)

[3.3 A program telepítése 6](#_Toc197278228)

[3.3.1 Asztali alkalmazás telepítése 6](#_Toc197278229)

[3.4 A Vendégház Vadaspark asztali alkalmazás használata 11](#_Toc197278230)

[3.4.1 1. Vendégeink menüpont 13](#_Toc197278231)

[3.4.2 Örökbefogadás menüpont 17](#_Toc197278232)

[3.4.3 Jegyrendelés menüpont 19](#_Toc197278233)

[3.4.4 Szerviz menüpont 20](#_Toc197278234)

[4 Fejlesztői dokumentáció 23](#_Toc197278235)

[4.1 Az alkalmazott fejlesztői eszközök 23](#_Toc197278236)

[4.2 Adatmodell leírása 24](#_Toc197278237)

[4.2.1 I. Adatbázis szerkezete 24](#_Toc197278238)

[4.2.2 II. Kapcsolatok az adattáblák között 26](#_Toc197278239)

[4.3 Részletes feladatspecifikáció, algoritmusok 28](#_Toc197278240)

[5 Tesztelési dokumentáció 28](#_Toc197278241)

[6 Összefoglalás 30](#_Toc197278242)

[6.1 Önértékelés 30](#_Toc197278243)

[6.2 Továbbfejlesztési lehetőségek 31](#_Toc197278244)

[7 Felhasznált irodalom 31](#_Toc197278245)

[8 Jogi nyilatkozat 32](#_Toc197278246)

[9 Ábrajegyzék 32](#_Toc197278247)

# Bevezetés

## Feladat leírás

A Vendégház Vadaspark projekt célja egy olyan komplex alkalmazás fejlesztése volt, amely egy vadaspark napi működését támogatja, mind webes, mind asztali alkalmazás formájában. A projekt címe: Vendégház Vadaspark.

A Vadasparkok jelenléte üde színfoltot jelent világszerte az embereknek a hétköznapi rohanásból való kilépésre, a kikapcsolódásra és feltöltődésre. A parkok látogatása minden korosztály számára rendkívül jó program, kikapcsolódás és tanulás szempontjából is. Akár egy egész napot is eltölthetünk itt, mert a park állatainak megtekintésén túl, az állat bemutatók, az élményetetések, a játszópark és étterem biztosítja kicsiknek és nagyoknak egyaránt a nap tartalmas eltöltését. S hogy ez megvalósuljon szükség van a háttérben folyó munkákra, melynek megvalósításában hivatott a projekt segítséget nyújtani.

A program használatához, telepítéséhez és beüzemeléséhez felhasználói dokumentáció készült, mely segítségével elsajátítható a projekt használata mindenki számára, aki használni kívánja azt. A dokumentáció lépésről lépésre mutatja be a program funkcióit, lehetőségeit.

Projekt célja, hogy egy könnyen kezelhető, naprakész, korszerű felhasználói felülettel rendelkező alkalmazás létrehozása, amely felhasználói számára lehetőséget nyújt az alábbi feladatok elvégzésére:

## A projekt két komponensből épül fel

### Asztali alkalmazás

A projekt asztali alkalmazásának feladata, hogy a Vadaspark munkatársai, munkájuk során több területen kezeljék és feldolgozzák a beérkező vagy változó adatokat.

Feladatok: Vendégeink, azaz a park lakóinak adatkezelése, feldolgozása, támogatói örökbefogadásuk intézése, weboldalon történt jegyfoglalás kezelése, illetve a munkatársak adatkezelése.

Fentiek alapján az alábbi feladatok ellátását az asztali alkalmazás teljesíti:

* Bejelentkezés - dolgozók részére
* Munkamenet választása
* Vendégeink
* Vendégeink oldalra, ahol választás alapján tovább léphetünk az
* Egyén oldalára, adataihoz
* Örökbefogadás
* Itt történik a támogatói örökbefogadás rögzítése, majd az
* Örökbefogadói oklevél készítése
* Jegyfoglalás
* A felhasználók által weboldalon történt jegyrendelés kezelése (módosítás, törlés)
* Szervíz
* Itt kezelhetjük a munkatársak adatait

Az asztali alkalmazást C# nyelven, WinForms keretrendszerben került fejlesztésre, amely stabil és egyszerű megoldást biztosít a Park munkatársainak napi munkájukhoz.

### Webes alkalmazás

A projekt webes alkalmazásának feladata, hogy a Vadaspark látogatói és felhasználói megismerhessék munkánkat, célkitűzéseinket, megismerhessék "vendégeinket", akiket támogatóként örökbe is fogadhatnak, segítséget nyújthatnak munkánkban, illetve jegyeket foglalhatnak a parkba történő látgatásukhoz.

Fentiek alapján az alábbi feladatok ellátását a web alkalmazás teljesíti:

* Bemutatkozás
* Vendégeink
* Megtekinthető állataink felsorolása alapján
* Örökbefogadás
* Támogatói örökbefogadás feltételeinek megtekintése, a már örökbefogadott állatok és örökbefogadóik listája látható itt
* Jegyrendelés
* Bejelentkezés
* Regisztráció

A webalkalmazás frontendje React, HTML, CSS, JavaScript technológiák felhasználásával készült, a backend alapokat Node.js és Express biztosítja.

## Miért ezt a projekt témát választottam?

Gyermekkorom óta életem szerves részét alkotja az állatok és az élővilág csodálata, szeretete. Nagyszüleim és szüleim is az állatok, növények tiszteletére és megbecsülésére tanítottak. Nem is volt kérdés, hogy témaválasztáskor erről a csodálatos területről választok!

A projekt készítésekor gyakorlatban alkalmazhattam a adatbázis-tervezés és kezelésben, felhasználói felület kialakításban, verziókezelésben és a csapatmunkában használhattam fel a megszerzett programozási tudásomat.

A projekt dokumentáció készítésekor cél volt, hogy olyan formában készüljön el, mely egyértelműen bemutatja a működését, illetve a felhasználókkal átláthatóan rmegismerteti minden szinten a program működését.

## A felhasznált ismeretek

A program elkészítésében során az **adatbázis-kezelés** ismeretei, **asztali alkalmazás fejlesztése**, a **backend és frontend programozás** területén megszerzett tudás segítette a munkát. Az iskolai tanulmányok mellett azonban szükség volt önálló tanulásra, fejlődésre ahhoz, hogy a projekt elkészülhessen.

A technológiai megvalósításhoz elengedhetetlen volt az alább részletezett felhasznált nyelvek és keretrendszerek ismerete.

* **C#** – az asztali alkalmazás elkészítéséhez
* **React** – a webes frontend modern, komponens-alapú kialakításához
* **Bootstrap** – a reszponzív, letisztult megjelenés megvalósításához
* **JavaScript, HTML, CSS** – a webes felülethez
* **Node.js és Express** – a szerveroldali működéshez és REST API-k létrehozásához
* **MySQL** – az adatbázis kezeléséhez, a kapcsolatok és lekérdezések kialakításához

Fenti ismereteket az iskolai oktatás, illetve különböző online forumok felhasználásával kerültek elsajátításra.

Fejlesztés közben alaposan meg kellett ismerkedni az **Express** keretrendszerrel és a **MySQL** adatbázis-motor működésével, a **GitHub-bal és a Gittel,** acsapatmunkához pedig elengedhetetlen volt a **Microsoft Teams. Ezen** területeken eddig megszerzett tudás és tapasztalatokat felhasználása nélkül a projekt nem készülhetett volna el.

## A felhasznált szoftverek

* **XAMPP** – a helyi szerver környezet biztosításához és a MySQL adatbázis futtatásához
* **phpMyAdmin** – az adatbázis kezelőfelületének használatához, amely segítséget nyújtott a táblák, kapcsolatok és lekérdezések vizuális kezelésében
* **Visual Studio Code** – a backend részének és a webes alkalmazás frontend fejlesztéséhez (React, Node.js, Express)
* **Visual Studio (C# WinForms)** – az asztali alkalmazás fejlesztéséhez

A fejlesztés folyamatosságát és stabilitását ezen szoftverek együttesen biztosították a munka során.

# Felhasználói dokumentáció

## A program általános specifikációja

A Vendégház Vadaspark projekt egy többplatformos alkalmazás, amely a Park munkáját segíti. Felhasználóbarát felületet biztosít a böngészéséhez, támogatói örökbefogadásához, jegyvásárláshoz és a park lakóinak adatkezeléséhez.

A program két modulból áll:

* **Asztali alkalmazás:** Feladata a park adminisztrációs feladatainak biztosítása, munkatársak és adminisztrátorok számára készült. Lehetővé teszi az adatbázisban tárolt adatok kezelését (új adat felvétele, meglévők szerkesztése, törlése, listázás). Kezeli az állatok-, az örökbefogadás jegyrendelés- és a szervíz feladatokat.
* **Webes alkalmazás:** Feladata annak biztosítása, hogy a látogatói és a felhasználói böngészhessenek vendégek- , a különböző programok és a szolgáltatások között, megismerkedhessenek a park mindennapjaival, -nyitva tartásával vagy éppen a támogatói örökbefogadásról érdeklődhessenek vagy éppen jegyet rendeljenek közelgő látogatásukhoz.

Az alkalmazás egyszerű használhatóságával, átlátható kezelő felületekkel a hatékonyságra törekszik, stabil háttértechnológiával és adatbázissal biztosítja a felhasználói élmény.

* **Rendszerkövetelmények**
* **Hardver követelmények**

Minimális követelmények:

* Processzor: Dual-core 1.8 GHz
* Memória: 4 GB RAM
* Tárhely: Legalább 500 MB szabad hely az alkalmazások és adatbázis számára
* Kijelző: Minimum 1366x768 felbontás
* Internetkapcsolat: Stabil, szélessávú internetkapcsolat webhez

Ajánlott konfiguráció:

* Processzor: Quad-core 2.5 GHz vagy gyorsabb
* Memória: 8 GB RAM vagy több
* Tárhely: Legalább 1 GB szabad hely
* Kijelző: Full HD (1920x1080) vagy nagyobb felbontás
* Internetkapcsolat: Stabil, szélessávú internetkapcsolat

Különleges hardverelemeket az alkalmazás nem igényel, a általában használatos otthoni vagy irodai gépeken futtatható.

## Szoftver követelmények

Adatbázis-kezeléshez:

* MySQL
* phpMyAdmin (XAMPP csomagon belül)

Asztali alkalmazás futtatásához:

* Operációs rendszer: Windows 10 vagy újabb
* Futtatókörnyezet: .NET Framework (ajánlott: .NET 4.7 vagy újabb)
* Fejlesztőkörnyezet (ha módosítani akarjuk a programot): Microsoft Visual Studio

Webes alkalmazás futtatásához:

* Operációs rendszer: Windows 10/11, MacOS vagy bármilyen Linux disztribúció
* Böngésző: Legfrissebb verziójú Google Chrome, Mozilla Firefox, Microsoft Edge vagy Safari
* Frontend futtatásához: React kompatibilis környezet
* Backend kiszolgálásához:
* Node.js (14-es verzió vagy újabb)
* Express.js keretrendszer
* MySQL szerver (XAMPP csomagon belül ajánlott)

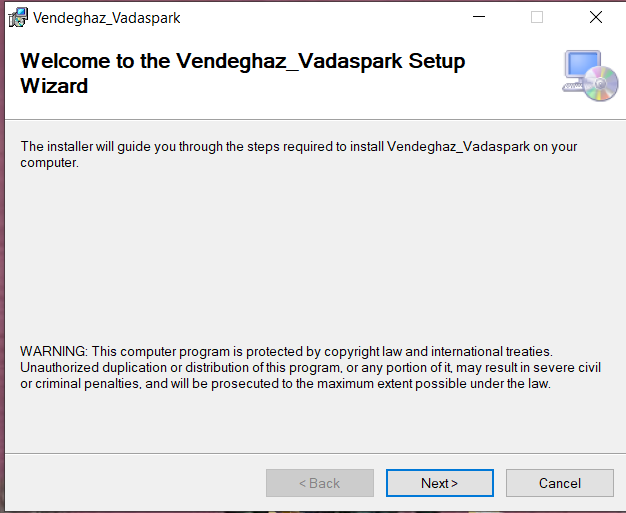
## A program telepítése

### Asztali alkalmazás telepítése

A telepítéshez szükséges ***’Vendeghaz\_Vadaspark.msi***’ fájl a vizsgaremek repository  
’Vendeghaz\Desktop\Vendeghaz\_Vadaspark\Release’  
 mappában található

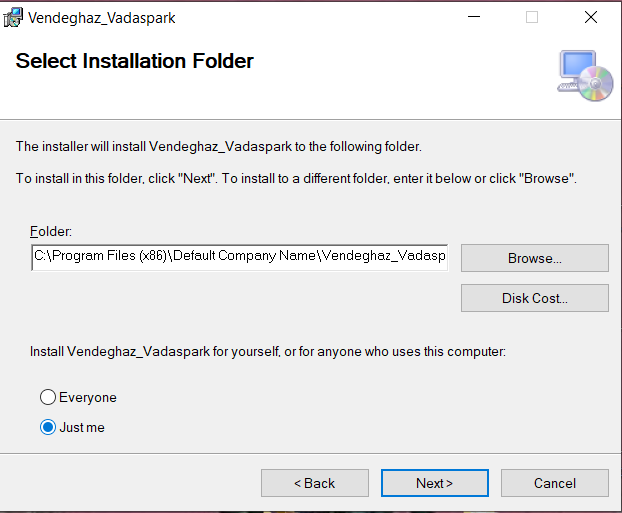
Ezt a fájlt futtatva elindul a telepítés.

A megjelenő ablakban:



1. ábra Telepítő ablak

a next gombra kell kattintani, ami a következő ablakba visz:



**2**. ábra Telepítő ablak 2

itt a BROWSE gomb segítségével felhasználó kiválaszthatja, hogy az alkalmazást hová szeretné telepíteni. Amennyiben az ajánlott útvonal megfelelő, abban az esetben a JUST ME bejelölést meghagyva ismét NEXT gomb megnyomásával tovább lépünk.



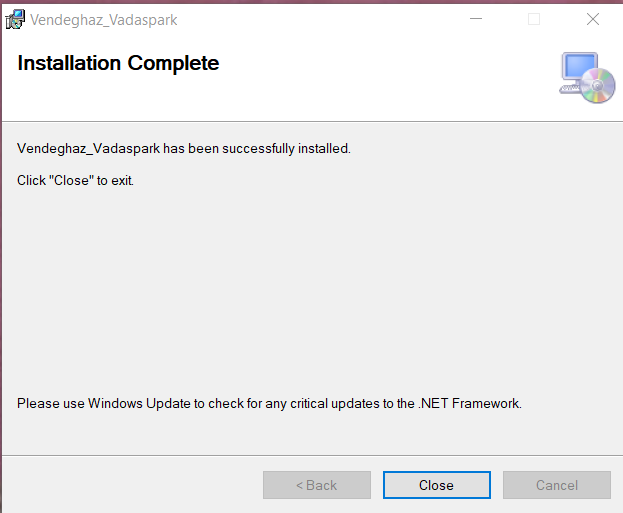
**3**. ábra Telepítő ablak 3

3.kép Telepítő ablak 3

a NEXT gomb lenyomása után már kezdődik is a telepítés.

A telepítés gyorsan lefut

és megjelenik az utolsó ablak a telepítésben:



**4**. ábra Telepítő ablak 4

melyet a CLOSE gomb megnyomásával fejezhetünk be.

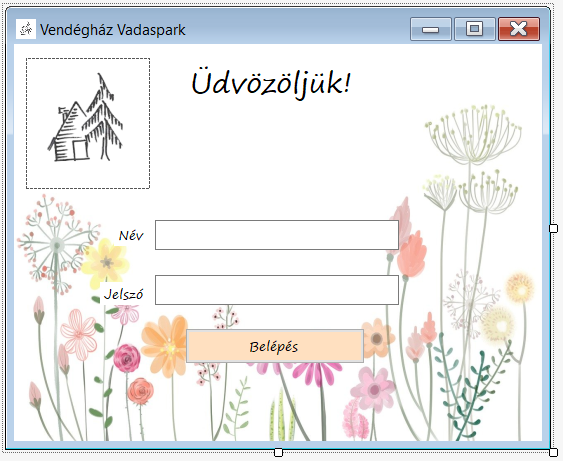
Ezután az alkalmazás Vendeghaz néven található meg a számítógépen:



Kezdő név és jelszópár: Teszt - Teszt123 jogosultság: admin

## A Vendégház Vadaspark asztali alkalmazás használata

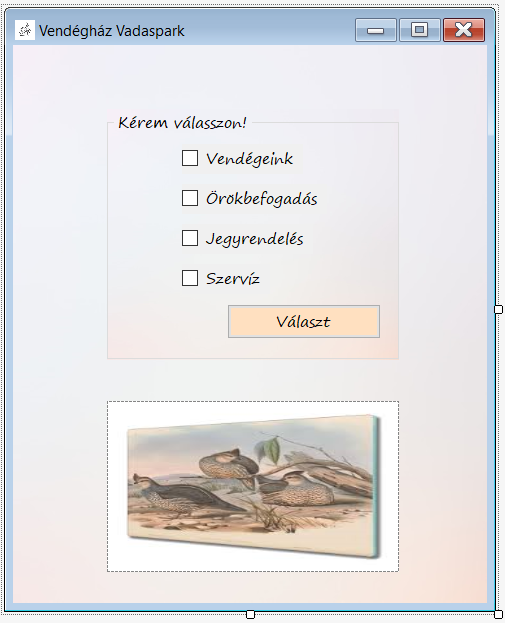
A munkafolyamat megkezdésének feltétele a sikeres belépés.



5. ábra Belépő ablak

Felhasználónak a név és jelszó megadása után van lehetősége belépni a prorgamba, hogy megkezdhesse munkáját. A név megadásának nincsenek feltételei, de a jelszónak minimum 3, maximum 10 karakter hosszúnak kell lennie, legalább 1 kis-, 1 nagybetűt és 1számot kell tartalmaznia. Megengedettek az ékezetes betűk használata.

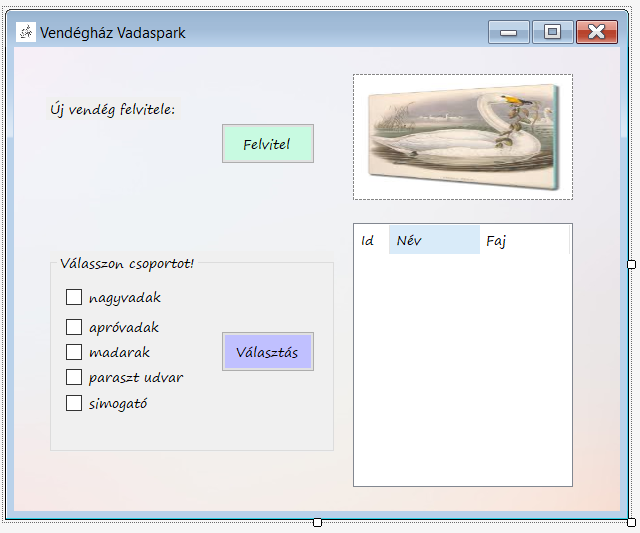
Sikeres bejelentkezés után a főmenűbe érkezünk:



6. ábra Főmenü

Itt az összes ellátandó terület közül választhatjuk ki a számunkra megfelelőt. A checkboxok segítségével kiválaszthatjuk mely területen szeretnénk dolgozni, majd a választásunkat agomb megnyomásával érvényesíthetjük.

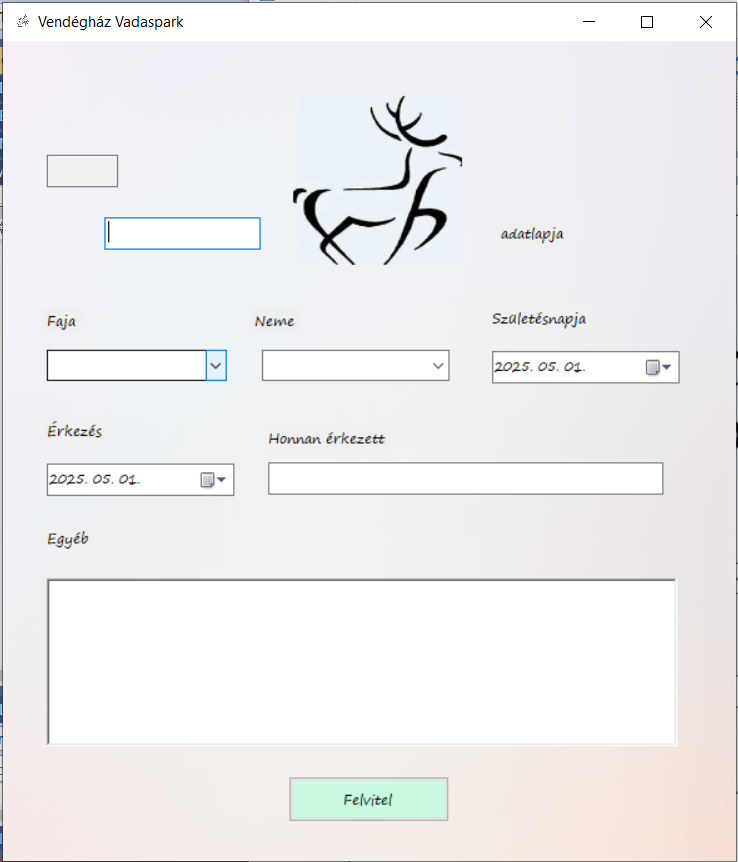
### 1. Vendégeink menüpont



7. ábra Vendégeink menüpont

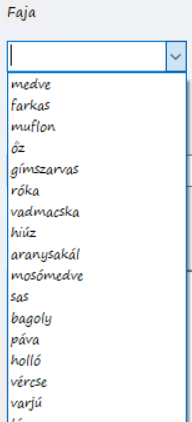
Ezen az oldalon a "vendégek" azaz a vadaspark lakóinak adatait kezeljük. Két irányban haladhatunk, első irány amikor teljesen új állat adatait szeretnénk rögzíteni.

Ekkor csak a  gomb lenyomására van szükségünk, s átlépünk a vendégek adatlap oldalára.

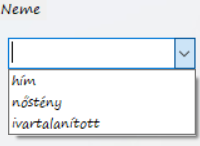


8. ábra Vendégeink felvitel adatlapja

Az első mezőben az új állatunk neve szerepel. A mellette lévő kép mezőben az állatkánk képe kerülhet, de ha még nincs róla fénykép üresen is hagyható. A képmező feltöltése képpel egy klikkel érhető el és a számítógép mappáiból kényelmesen kiválasztható a kívánt kép. A kép adatbázisba történő elmentésekor az állat nevét kapja automatikusan, tehát a kép átnevezésével nem kell foglalatoskodnunk, csak helyesen kell kiválasztanunk magát a képet. A következő mező a faj kiválasztást-, míg az azután következő a nem választást teszi lehetővé legördülő menük alapján. Kiválasztás klikkel történik.



**9**. ábra Faj és nem választás I.



10. ábra Faj és nem választás II.

Következik a születési dátum és a parkba érkezés dátumainak beállítása, mely könnyen megtehető a naptár funkció  használatával. Következik a "Honnan érkezett" mező, itt a parkba érkezés dátuma mellett a helyet rögzítjük, ahonnan az adott állunk érkezett. Legvégül a egyéb mezőben a parklapónkkal kapcsolatos feljegyzéseket lehet megtenni. Ezen mező kitöltése nem kötelező.

Amennyiben a Vendégek menüpont másik irányában haladunk tovább, akkor a már meglévő adatlapokban lesz lehetőségünk módosítani vagy törölni. De előtte ki kell választanunk az egyedet. A választáshoz egy csoportosítást használunk, mely élettani vagy területi csoportosításban sorolja fel az állat fajtákat.

Nagyvadak:

'medve','farkas','muflon','őz','gímszarvas', 'vaddisznó'

Apróvadak:

'róka','vadmacska','hiúz','aranysakál','mosómedve'

Madarak:

'sas','bagoly','páva','holló','vércse','varjú'

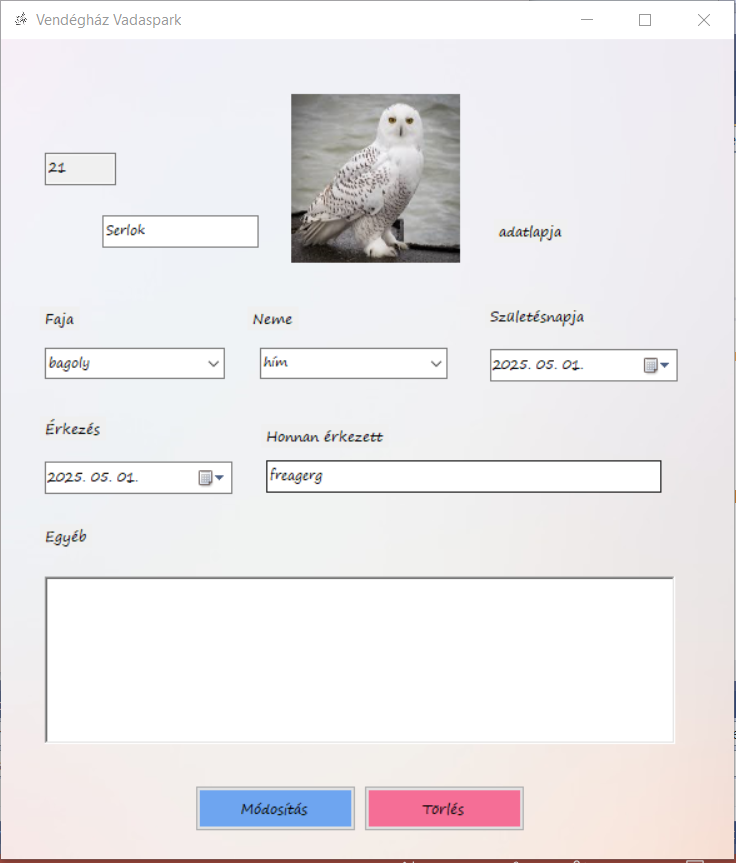
Parasztudvar:

'ló','szamár','tehén','mangalica','baromfiak'

Simogató:

'dámszarvas','juh','kecske','nyúl','póniló'

Checkbox választással szűkíthetjük a keresést, majd a  gombbal a listában megjelent állatok közül egér klikkel választva léphetünk át a már felvitelnél megismert adatlap oldalra, mely ekkor már az egyed minden adatával feltöltve jelenik meg.



11. ábra Vendégeink módosítás és törlés adatlapja

Itt a felvitelhez hasonlóan tudjuk módosítani a mezőket vagy éppen cserélhetünk képet is, de itt törölhetjük az adatlapot is.

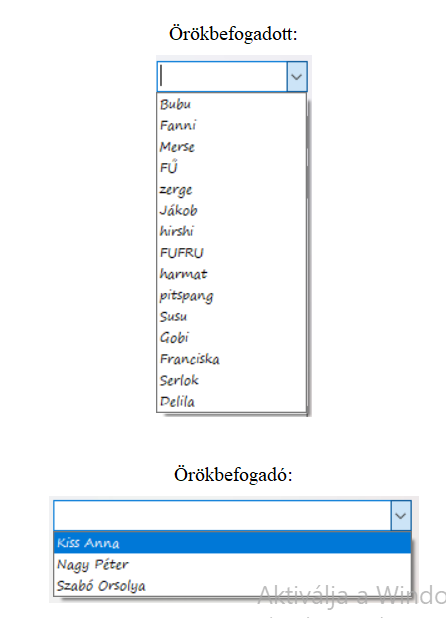
### Örökbefogadás menüpont

A támogatói örökbefogadás egyfajta adományozás, mellyel a park mindennapi életét lehet támogatni. Az örökbefogadó egy éven keresztül viseli az örökbefogadó szülő címet.



12. ábra Örökbefogadás

Ezen az adatlapon két legördülő menüben választhatjuk ki az örökbefogadott és az örökbefogadó személye.



13. ábra Örökbefogadás választások

Ha kiválasztásra kerül mindkét fél, akkor a  gomb megnyomásával már rögzítésre is kerül az örökbefogadás.

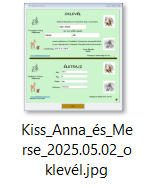
Ha már választottunk, de mégsem szeretnénk az rögzíteni, akkor az  gomb megnyomásával törlődnek a feltöltött adatok, folytatódhat az újbóli kiválasztás.

Az örökbefogadás rögzítésén felül, a köszönetnyilvánítás jeléül az örökbefogadó szülő egy oklevelet kap és egyben egy éven keresztül viseli a megtisztelő örökbefogadó szülő címet. Ha a rögzítést választottuk, akkor az oklevél oldalra jutunk, ahol eldönthetjük, hogy az oklevelet .jpg formában mentjük-e vagy csak kilépünk.



14. ábra Oklevél

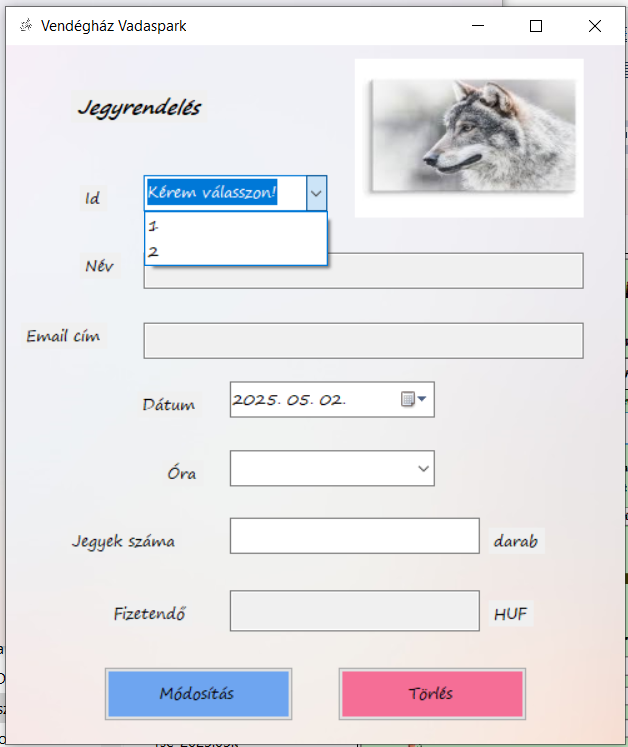
Mentés alkalmával a jpg az örökbefogadó, az örökbefogadott és a dátum kombinációját kapja megnevezésnek (*pl: Kiss\_Anna\_és\_Merse\_2025.05.02\_oklevél.jpg*), mely a .\bin\Debug\Contract mappában kerül mentésre.



. ábra oklevél JPG

### Jegyrendelés menüpont

A jegyrendelés menüben a felhasználók által a weboldalon rögzített jegyrendeléseket lehet módosítani és törölni.



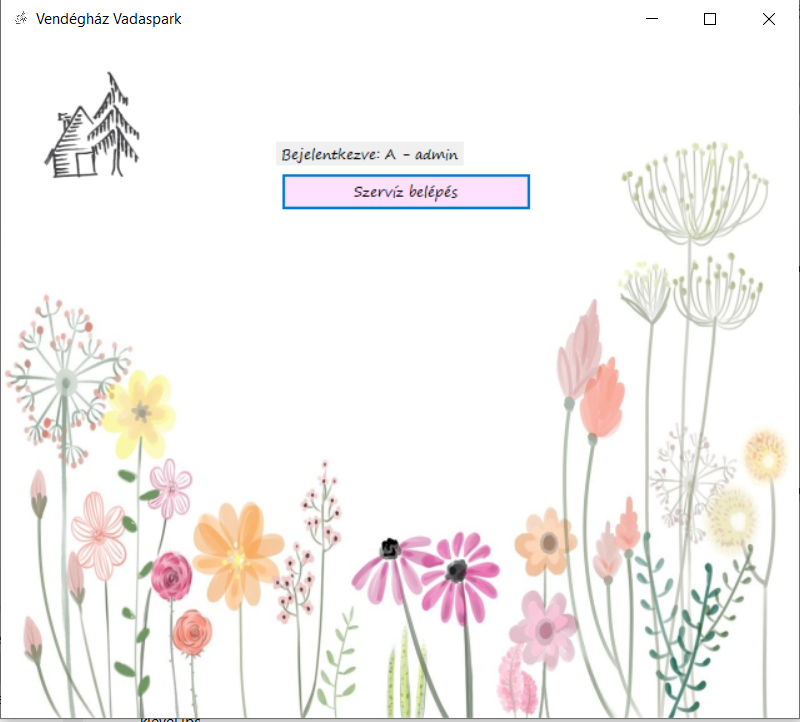
16. ábra Jegyrendelés adatlapja

Az oldalon a betöltődött jegyrendelés számok alapján kérhetjük le a jegyek adatait. Egér klikk választás után feltöltődnek az adatok. Itt a dátum, óra és a jegyek száma mezőkben

 gombbal módosíthatunk vagy  gombbal törölhetjük a rendelést.

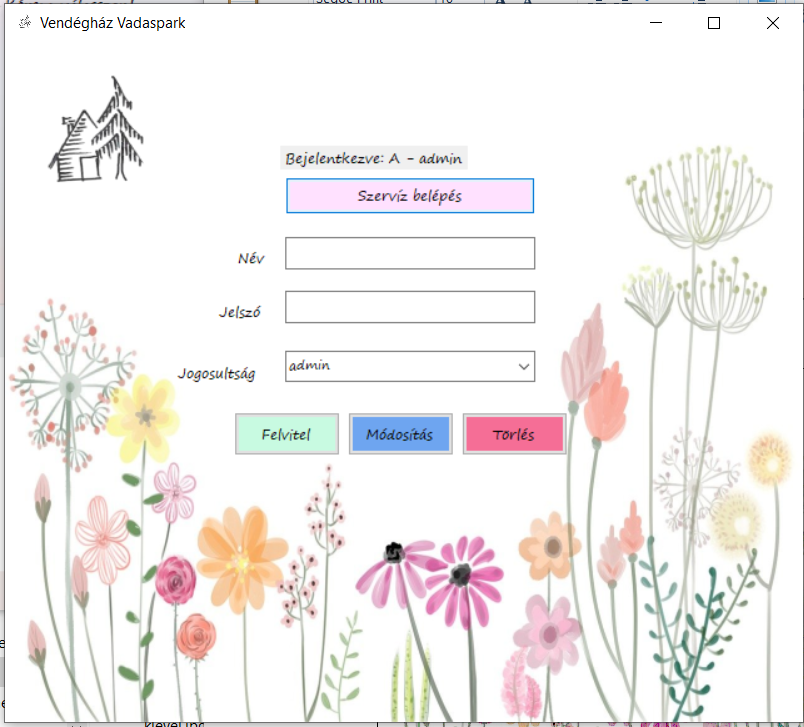
### Szerviz menüpont

Szervíz menü a munkatársak adatainak kezelésére szolgál. Felvihetünk, módosíthatunk vagy törölhetünk személyeket, de csak abban az esetben, ha a programba bejelentkezett személy admin jogosultságú. Az adatlap  gombja lenyomás után ellenőrzi, hogy csak arra jogosult személy léphessen tovább a funkciók eléréséhez.



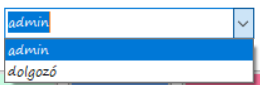
17. ábra Szervíz adatlap belépés

Ha jogosult személy van bejelentkezve, akkor lehetősége nyílik felvitelre, módosításra vagy törlésre.



18. ábra Szervíz adatlap

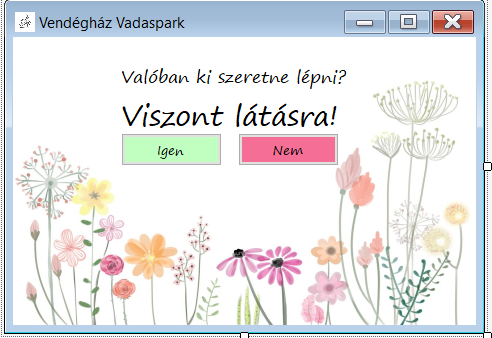
A név és jelszó kitöltés szabályai megegyeznek a belépéskor ellenőrzött elemekkel: a név megadásának nincsenek feltételei, de a jelszónak minimum 3, maximum 10 karakter hosszúnak kell lennie, legalább 1 kis-, 1 nagybetűt és 1számot kell tartalmaznia. Megengedettek az ékezetes betűk használata. A legördülő menüben választhatjuk ki a jogosultságot.



19. ábra Szervíz jogosultság választás

Miután kitöltöttük a mezőket a  gombokkal végezhetjük el a megfelelő műveletet. A törlés gomb használatakor nem szükséges minden mező kitöltésénél, elég ha csak a nevet írjuk be.

Minden munkafolyamat a főmenüre lép vissza, ezért a munka befejezése csak itt lehetséges! Amennyiben szeretnénk a programot bezárni azt itt a **bezárás** () gomb megnyomásával tehetjük meg. Innen a kilépés megerősítésével vagy éppen az elvetésével haladhatunk tovább.



20. ábra Kilépés

Az gomb választásakor a program bezáródik és leáll, míg a  használata után a főmenübe folytathatjuk ismét a munkát.

# Fejlesztői dokumentáció

## Az alkalmazott fejlesztői eszközök

A fejlesztés során használt nyelvek és eszközök mindegyike nyílt forráskódú, ingyenesen elérhető vagy oktatási célra jogszerűen használhatóak.

Programozási nyelvek:

* SQL – Az adatbázis műveletekhez (lekérdezések, adatmanipulációk)
* C# – A WinForms-alapú asztali alkalmazás megvalósításához
* JavaScript – A webes alkalmazás dinamikus működéséhez
* HTML és CSS – A felhasználói felület szerkezetének és stílusának kialakításához

Fejlesztői környezetek és platformok:

* XAMPP – A MySQL adatbázis és a phpMyAdmin kezelőfelület helyi futtatására
* phpMyAdmin – Az adatbázis grafikus kezelőfelülete
* Visual Studio – A C# alapú asztali alkalmazás fejlesztésére
* Visual Studio Code – A frontend és backend fejlesztéshez (React, Node.js, Express)

Backend és adatbázis:

* Node.js + Express.js – A REST API-k fejlesztéséhez és szerveroldali logika megvalósításához
* MySQL – Az alkalmazás adatainak tárolásához és kezeléséhez

Frontend:

* React – A webes felület komponensalapú felépítéséhez
* Bootstrap – A reszponzív megjelenés biztosításához (nyílt forráskódú CSS keretrendszer)
* Visual Studio (C# WinForms) – az asztali alkalmazás fejlesztéséhez

Verziókezelés és csapatmunka:

* Git + GitHub – Verziókezelés és kollaborációs feladatokhoz
* Microsoft Teams

Dokumentáció készítéséhez használt szoftverek:

* Microsoft Word / Google Docs – Dokumentáció megírására
* Greenshot / Lightshot – Képernyőképek készítéséhez, illusztráláshoz a működés bemutatására
* Paint – Egyszerű képszerkesztési feladatokhoz

Külső modulok és csomagok:

* MySQL2 (npm csomag) – A Node.js és MySQL közötti kapcsolat biztosítására
* React Router – A frontend oldali navigáció kezeléséhez

A használt külső könyvtárak és csomagok mind nyílt forráskódúak, és a hivatalos npm registry-n keresztül letölthetők, így jogszerűen alkalmazhatók oktatási célú projektekben.

## Adatmodell leírása

Az adatok kezelésére a projekt egy relációs adatbázist használ, minden entitást és kapcsolataikat tartalmazza mely a működéshez szükséges.

### I. Adatbázis szerkezete

**Az adatbázis neve: vendeghaz**

Célja a Vadaspark adatainak hatékony és biztonságos tárolása.

Az adatbázis az alábbi fő adattáblákat tartalmazza:

**1. guests (Állatok/Vendégek)**

Mezőnév Típus Leírás

g\_id BIGINT UNSIGNED PRIMARY KEY Egyedi azonosító

g\_name VARCHAR(50) Állat neve

g\_species ENUM Faj (pl.: medve, farkas, őz stb.)

g\_gender ENUM Nem (hím, nőstény, ivartalanított)

g\_birthdate DATE Születési dátum

g\_indate DATE Befogadás dátuma

g\_inplace VARCHAR(100) Befogadás helye

g\_other TEXT Egyéb megjegyzés

g\_image VARCHAR(255) Állat képének fájlneve

2. users (Felhasználók)

Mezőnév Típus Leírás

u\_id BIGINT UNSIGNED PRIMARY KEY Egyedi azonosító

u\_name VARCHAR(255) Felhasználó neve

u\_email VARCHAR(255) Felhasználó e-mail címe

u\_phone VARCHAR(20) Telefonszám

u\_address VARCHAR(255) Lakcím

3. workers (Dolgozók)

Mezőnév Típus Leírás

w\_id BIGINT UNSIGNED PRIMARY KEY Egyedi azonosító

w\_name VARCHAR(255) Dolgozó neve

w\_password VARCHAR(255) Jelszó (titkosított)

w\_role ENUM ('admin', 'user') Dolgozó szerepköre

4. adoption (Örökbefogadás)

Mezőnév Típus Leírás

a\_id BIGINT UNSIGNED PRIMARY KEY Egyedi örökbefogadás azonosító

g\_id BIGINT UNSIGNED FOREIGN KEY (guests.g\_id) Örökbefogadott állat azonosító

u\_id BIGINT UNSIGNED FOREIGN KEY (users.u\_id) Örökbefogadó felhasználó azonosító

a\_date DATE Örökbefogadás dátuma

5. tickets (Jegyek)

Mezőnév Típus Leírás

t\_id BIGINT UNSIGNED PRIMARY KEY Jegy azonosító

t\_date DATE Jegy kiállításának dátuma

t\_type ENUM('felnőtt', 'gyerek', 'családi') Jegytípus

t\_price DECIMAL(10,2) Jegy ára

### II. Kapcsolatok az adattáblák között

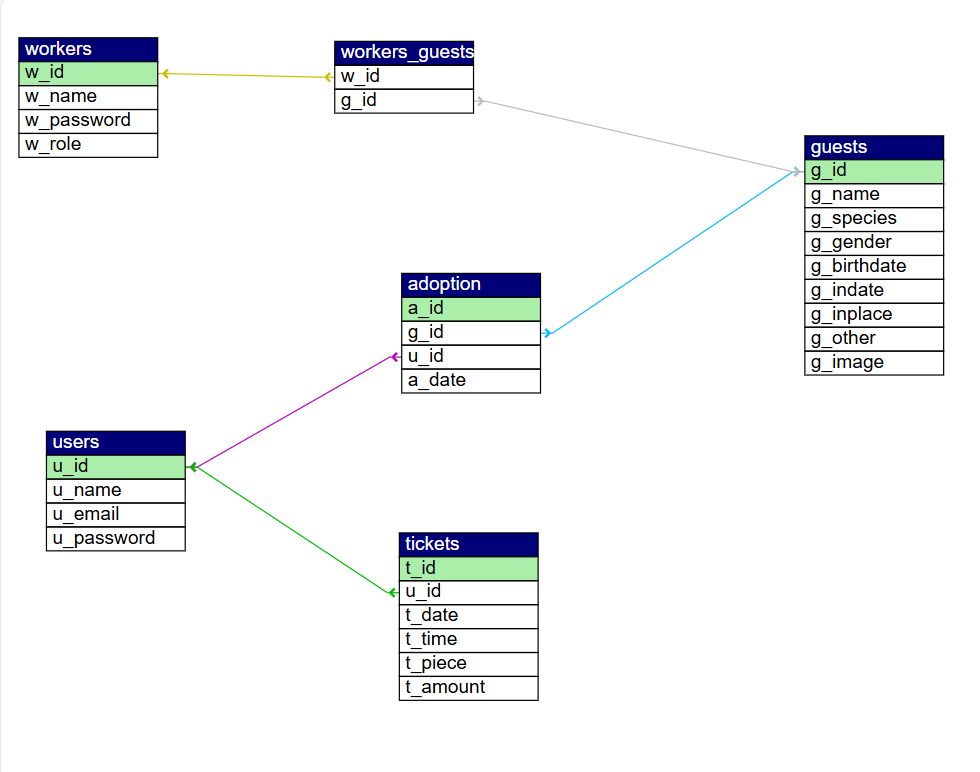
1. A guests és adoption táblák között: 1:N kapcsolat

(Egy állatot többször is örökbe fogadhatnak - pl. támogatás formájában.)

2. A users és adoption táblák között: 1:N kapcsolat

(Egy felhasználó több állatot is örökbe fogadhat.)

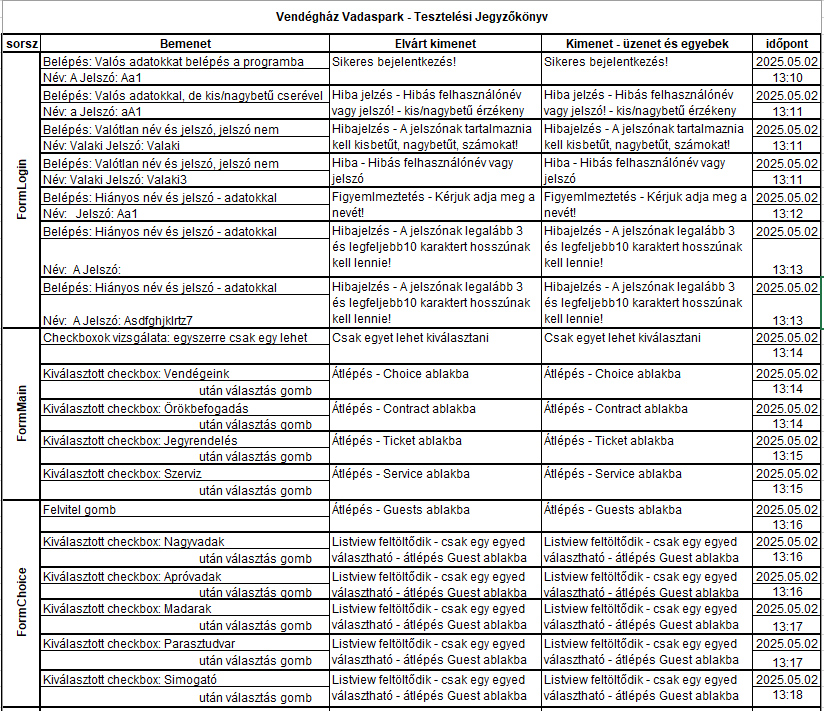
A workers tábla nincs közvetlen kapcsolatban más táblákkal – ők a háttérrendszer kezelését végzik.

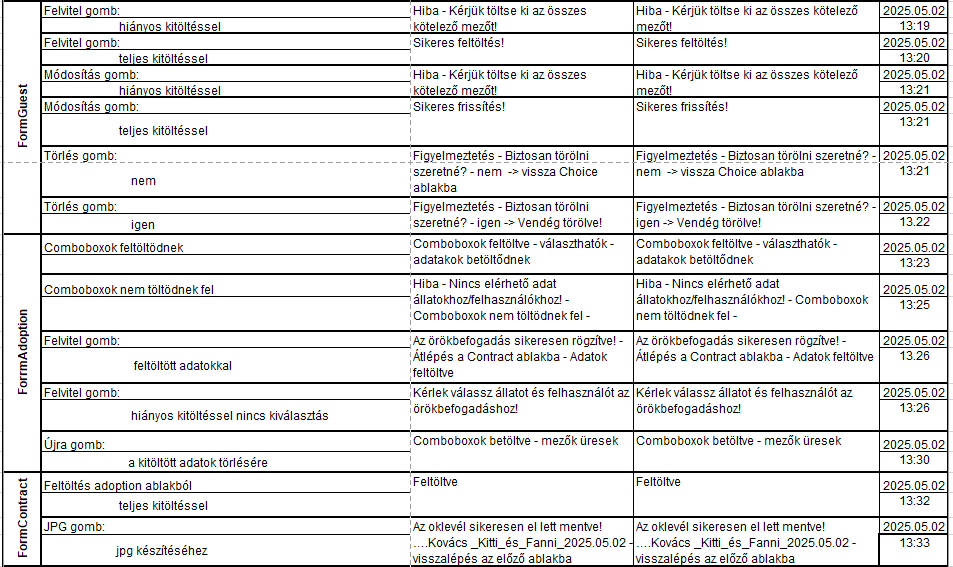


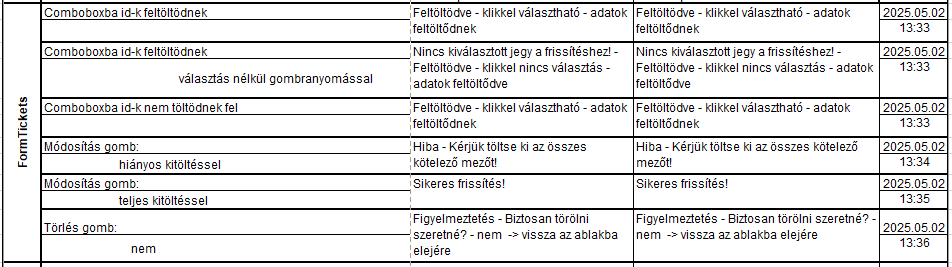
21. ábra Adatbázis ER diagram

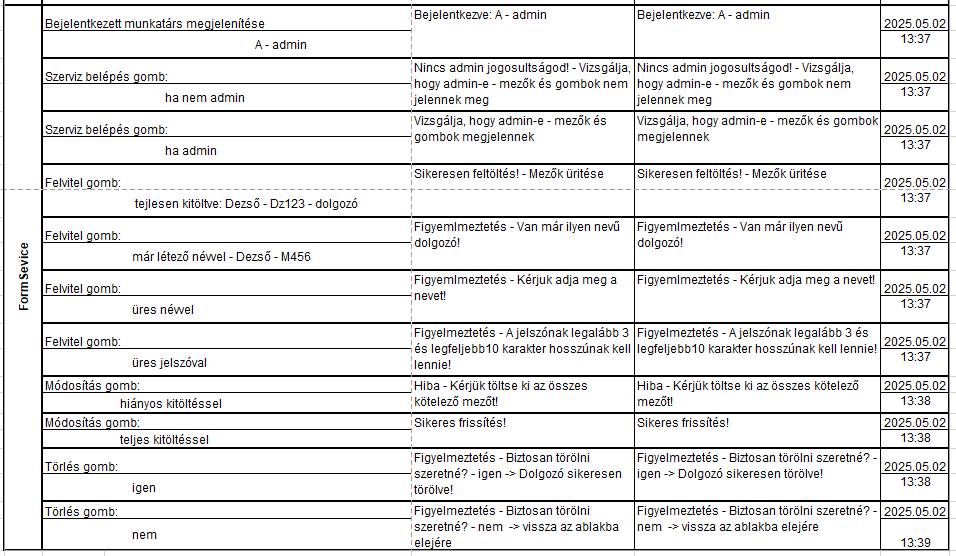
## Részletes feladatspecifikáció, algoritmusok

# Tesztelési dokumentáció









# Összefoglalás

## Önértékelés

A projekt létrehozásakor kitűzött cél volt, hogy minden szempontból működőképes és átlátható webes és asztali alkalmazás legyen, ezáltal biztosítsa egy felhasználóbarát felület létre jöttét, amely lehetőséget nyújt mindenki számára betekintést nyújtani a park minden napijaiba, programjaiba és lakóinak életébe.

A készítés során számos technikai nehézség állt a fejlesztés útjában, mivel tanulmányaink során ez volt az első igazán komplex és nagy volumenű projekt, melynek megvalósítása az elejétől a végéig a mi felelősségünk volt. Alapos átgondolást igényelt a teljes projekt megtervezése. Törekeni kellett a folyamat során arra, hogy minden a megfelelő sorrendben kerüljön megvalósításra, hogy elkerüljük a plusz munkát vagy esetleges ütközéseket.

A felmerült problémák és azokon való átlendülés hozzájárultak a fejlődéshez. A kihívások között voltak például a megfelelő adatbázis struktúra létrehozása, mely megfelelő alapot szolgált a backend és frontend közötti megfelelő kapcsolathoz. Ennek kialakítása során újabb és újabb dolgokat kellett elsajátítani, sőt kísérletekgetni is, hogy az eltervezett cél megvalósulhasson.

## Továbbfejlesztési lehetőségek

Egy vadaspark napi élete sokrétű feladatokat foglal magába. Az üzemeltetés és a park lakóinak ellátása mellett az ide látogató emberek kiszolgálása és az egész napos látnivalókkal való ellátása is nagy és összetett feladat. A projekt ezért több vonalon is tovább fejleszthető.

Jövőben megvalósítható fejlesztés:

A projekt további fejlesztését az asztali alkalmazásban a park lakóinak egészségügyi kartonjának megvalósításával kezdeném, mely nagy segítséget nyújtana a park munkatársai számára a napi rutin során. Így a nap kezdetével már látható lenne az egyes orvosi teendők listája (pl: orvosi vizit, gyógyszerezés, időszakos megelőző kezelések stb.). Betekinthetnénk az egyedek egészségügyi kartonjába is, látnánk eddigi kezeléseiket vagy éppen az egészségügyi állapotukat.

További porjekt bővítésként a beszerzés/készlet oldalon folytatnám a fejlesztést. Itt ketté osztanám a beszerzés vonalát: egyrészt a park lakókkal, másrészről az üzemeltetéssel kapcsolatos feladatok ellátásával folytatnám a projekt bővítést.

Webes felületen a felhasznlói visszajelzések megvalósításával bővíteném a feladatokat, hogy a beérkezett vélemények felhasználásával a park az igények alapján tudja alakítani munkáját, szolgáltatásait. Továbbá az oldalon történő jegyvásárlás fizetési integrációja és az események autómatikus értesítése felhasználók részére is egy-egy kitűzött cél lehetne.

Meg nem valósult fejlesztés:

A web oldalon időhiány miatt nem sikerült implementálni a jegyrendelés funkciót, illetve a bejelentkezés és regisztráció sem működik rendesen.

# Felhasznált irodalom

A projekt fejlesztése során elsősorban hivatalos dokumentációk, szakmai útmutatók és megbízható közösségi források felhasználása történt, amely nagyban hozzájárult a problémák hatékony megoldásához, a technológiák mélyebb megértéséhez és a rendszer sikeres megvalósításához.

* **Microsoft Docs – C# Windows Forms** <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/> A Windows Forms alkalmazások fejlesztéséhez és az űrlapvezérlők kezeléséhez.
* **.NET Documentation** <https://learn.microsoft.com/en-us/dotnet/> A .NET környezet, aszinkron programozás (async/await) és hálózati kommunikáció (HttpClient) használatához.
* **Node.js Documentation** <https://nodejs.org/en/docs/> A backend szerver felépítéséhez szükséges Node.js működésének megértéséhez.
* Express.js Guide <https://expressjs.com/> Az Express keretrendszer használatához, API-k fejlesztéséhez.
* MySQL Documentation <https://dev.mysql.com/doc/> Az adatbázis-kezeléshez, SQL lekérdezések írásához és adatstruktúrák kezeléséhez.
* Git Documentation <https://git-scm.com/doc> Verziókezeléshez, lokális munkamenetek kezeléséhez.
* GitHub Docs <https://docs.github.com/en> Az online tárolás, közös munkavégzés, pull requestek és verziókövetés használatához.
* REST API Concepts and Best Practices <https://restfulapi.net/> REST API működésének megértéséhez, helyes szerver-kliens kommunikáció kialakításához.
* Stack Overflow <https://stackoverflow.com/> Különböző fejlesztési problémák megoldásához, példák és közösségi tanácsok kereséséhez.
* W3Schools – Web Development Tutorials <https://www.w3schools.com/> Alapvető webfejlesztési tudnivalók áttekintéséhez (HTML, CSS, JavaScript, HTTP alapok).
* GeeksforGeeks <https://www.geeksforgeeks.org/> Általános programozási elvek, adatbázis-kezelés és szerver-oldali technológiák megismeréséhez.
* YouTube Programozási Oktatóvideók <https://www.youtube.com/> Gyakorlati példák, tutorialok és fejlesztési folyamatok megértéséhez különböző programozási nyelveken és technológiákban.

# Jogi nyilatkozat

A munka során felhasznált képek ingyenes képletöltő oldalakról származnak. **A Budakeszi Vadaspark engedélyével** a weboldalukon szereplő dokumentáció és képanyag részben felhasználásra került.

# Ábrajegyzék

[1. ábra Telepítő ablak 7](#_Toc197278366)

[**2**. ábra Telepítő ablak 2 8](#_Toc197278367)

[**3**. ábra Telepítő ablak 3 9](#_Toc197278368)

[**4**. ábra Telepítő ablak 4 10](#_Toc197278369)

[5. ábra Belépő ablak 11](#_Toc197278370)

[6. ábra Főmenü 12](#_Toc197278371)

[7. ábra Vendégeink menüpont 13](#_Toc197278372)

[8. ábra Vendégeink felvitel adatlapja 14](#_Toc197278373)

[**9**. ábra Faj és nem választás I. 15](#_Toc197278374)

[10. ábra Faj és nem választás II. 15](#_Toc197278375)

[11. ábra Vendégeink módosítás és törlés adatlapja 16](#_Toc197278376)

[12. ábra Örökbefogadás 17](#_Toc197278377)

[13. ábra Örökbefogadás választások 18](#_Toc197278378)

[14. ábra Oklevél 19](#_Toc197278379)

[15. ábra oklevél JPG 19](#_Toc197278380)

[16. ábra Jegyrendelés adatlapja 20](#_Toc197278381)

[17. ábra Szervíz adatlap belépés 21](#_Toc197278382)

[18. ábra Szervíz adatlap 22](#_Toc197278383)

[19. ábra Szervíz jogosultság választás 22](#_Toc197278384)

[20. ábra Kilépés 23](#_Toc197278385)

[21. ábra Adatbázis ER diagram 27](#_Toc197278386)