

Tolna Vármegyei SZC Apáczai
Csere János Technikum és Kollégium

Az ágazat megnevezése: Informatika és távközlés

A szakma megnevezése: Szoftverfejlesztő és -tesztelő

A szakma azonosító száma: 5 0613 12 03

Mancs Társ

Készítette

Szili Bettina

Rákóczi Viktória

Petrás Kata Magdolna

2/14 Sz-V osztály tanulói.

Dátum: 2025.

Feladat rövid ismertetése	3
Bevezetés.....	3
Cél.....	3
Használt technológiák	3
Programozási nyelvek.....	3
Adatbázis	4
Entity-relációs diagram:.....	4
Egyéb technológia.....	4
Adatbázis	5
Adatbázis rövid magyarázat	5
Fejlesztési módszertan	5
Scrum	5
Program működésének a részletes leírása	6
Program részletes leírása	6
Tesztelés	7
Tesztelési fajták	7
Jelentőség	7
Használt technológia.....	7
Frontend	7
Backend.....	7
Összegzés és továbbfejlesztési lehetőségek	8
Tapasztalatok	8
Továbbfejlesztési lehetőségek	8
Telepítési útmutató	12
Használt programok:	12
Használt program nyelvek:.....	12
Használt keretrendszerek:	12
Frontendhez használt programok letöltése és telepítése	16
Manuális Tesztelés	19

Feladat rövid ismertetése

Bevezetés

A MancsTárs menhelyen élő állatoknak segít szerető otthonra találni azzal, hogy ad a menhelyeknek egy nagyobb platformot, aminek segítségével minél több potenciális gazdát tudnak elérni, hiszen sokszor az vehető észre, hogy szinte alig találhatók meg online, illetve ha mégis, akkor is csak nehézkesen. A lehetséges gazdáknak pedig sokkal könnyebb lenne, hogyha minden menhely megtalálható lenne egy helyen, hiszen egy örökbefogadási rendszer elég körülményes tud lenni, mivel sokszor rengeteget kell utazniuk, hogy meglátogassanak egy menhelyet, hiszen nem mindig található egy a közelben és van, hogy néha többet is meg kell látogatniuk. Sokszor az is észrevehető, hogy az emberek és az állatok szükséglete mindezek után nem találkozik, ami ahhoz is vezethet, hogy az állatokat visszaviszik a menhelyre. A Mancstárs ezekre a problémákra próbál megoldást találni.

Cél

MancsTárs nevezetű projektnek az a célja a menhelyen lévő állatok tudjanak találni maguknak egy megfelelő otthont, ahol gondoskodnak róluk és nem hagyják el őket. Az oldalon minden állathoz tartozik egy kártya, ami olyan információkat tartalmaz róluk, amik könnyebben elősegítik azt, hogy a hozzájuk legjobban illő személyhez kerüljenek, hiszen többek között a fizika tulajdonságaik mellett, a személyiségükről is található leírás, egy pontozási rendszer alapján, így a jövőbeli örökbe fogadó könnyen kitudja választani azt a kisállatot, ami számára a legmegfelelőbb. Ezzel elkerülve azt, hogy a kisállatok és az emberek is csalódással zárják az örökbefogadási folyamatot.

Használt technológiák

Programozási nyelvek

Frontend-keretrendszerként az Angular 19-et és a Bootstrap-et használtuk, TypeScript nyelven, illetve az Angular komponenseinek és nézeteinek a kialakítását HTML-lel és CSS-el készítettük el.

Backend fejlesztéshez és az objektumorientált programozáshoz: A Java programnyelvet választottuk, mert számunkra ez a program nyelv sokkal kényelmesebbnek bizonyult.

Adatbázis-kezelésre a MySQL-t használtuk.

REST API-k használatával kérjük le és jelenítjük meg az adatbázisból érkező állatok és menhelyek adatait.

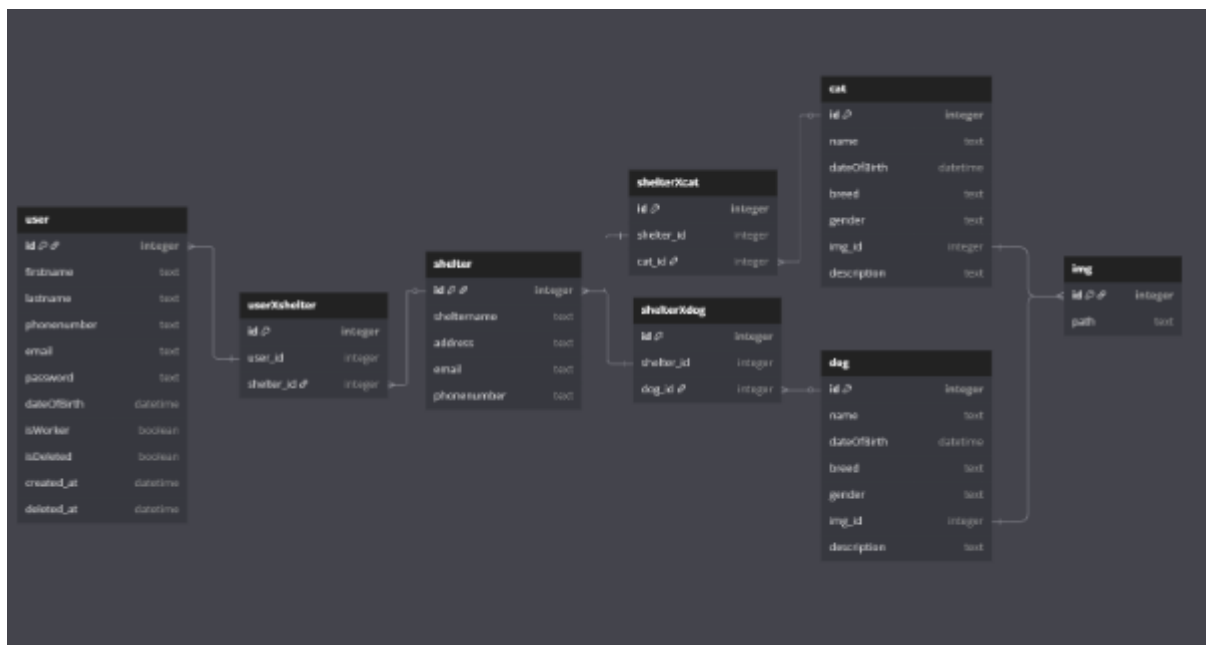
JWT (JSON Web Token) alapú autentikáció segítségével védjük le a felhasználóink személyes adatait, mert a felhasználói biztonság és elégedettség a két fő prioritásunk.

Fejlesztői környezetek, illetve a csomagkezelők pedig az alábbiak voltak: Visual Studio Code, Node.js és Netbeans.

Adatbázis

Relációs adatbázist használunk, amelyet a MySQL kezel, aminek segítségével könnyebben tudjuk az adatokat táblába rendezni, definiálni a köztük lévő kapcsolatot, valamint az adatok lekérdezése is hatékonyabb, a hozzá tartozó szervert pedig a MAMP segítségével futtatjuk.

Entity-relációs diagram:



Egyéb technológia

Postman-t és WildFly-t használtunk a REST API teszteléséhez és futtatásához, Bootstrap-et az oldal reszponzivitásának megvalósításához.

Csapatmunkához, legfőképp a Jira-t használtuk, verzió kezeléshez pedig GitHubot használtuk, továbbá az egymással való kommunikációra és a dolgok megosztásához a Google Drive-ot és a Discord-ot vettük igénybe.

A projektünknek a design terve Figmában készült el.

Adatbázis

Adatbázis rövid magyarázat

Adatbázisba először feltöltöttük adatokkal a kutya, macska, felhasználó, menhely táblát utána írtunk hozzá eljárásokat, amik lekérlik az összes kutya, macska, felhasználó és menhely adatait, utána megcsináltuk, hogy új adatokat tudjanak felvenni és ki is tudjanak törölni az álatokkal kapcsolatban az adminok, illetve frissíteni az adatokat. Új jelszó kérést is csináltunk, illetve regisztrálni és bejelentkezni is lehet.

Fejlesztési módszertan

GitHub repository: <https://github.com/szilibettina06/MancsTars>

Scrum

A fejlesztés során az Agil megközelítést alkalmaztuk, ezen belül is a Scrum módszertant. Ez tette lehetővé, hogy rugalmas módon haladjunk előre.

A csapatunk három főből állt, és a frontend, backend, adatbázis és a dokumentációs feladatokat osztottuk fel egymás között, még pedig úgy, hogy mindenki vállalta azt, amihez értett vagy fejlődni szeretett volna benne.

A fejlesztést havi sprintekre bontottuk, Minden hónap elején összeállítottuk a sprint célkitűzéseit, meghatároztuk, hogy ilyen feladatokat kell előtérben helyeznünk.

A feladatokat a Jira-ban vezettük, itt hoztuk létre a feladatokat, határidőket állítottunk be, jelöltük a feladatok aktuális állapotát, és követtük a haladásunkat A Jira átláthatóvá tette a csapatunk számára, hogy ki min dolgozott éppen, mi van még hátra és mi készült el.

Habár nem mindennap tartottunk megbeszéléseket, hetente legalább egyszer röviden átbeszéltük, hogy ki mivel haladt, milyen akadályba ütközött ilyen feladatokra fog fókuszálni az elkövetkezendő napokban.

A forráskódot a GitHub segítségével kezeltük, így hatékonyan tudtunk együttműködni hiszen mindenki külön ágon tudott dolgozni, majd a legvégén pull request-tel mindent sikeresen átintegráltunk a fő ágba.

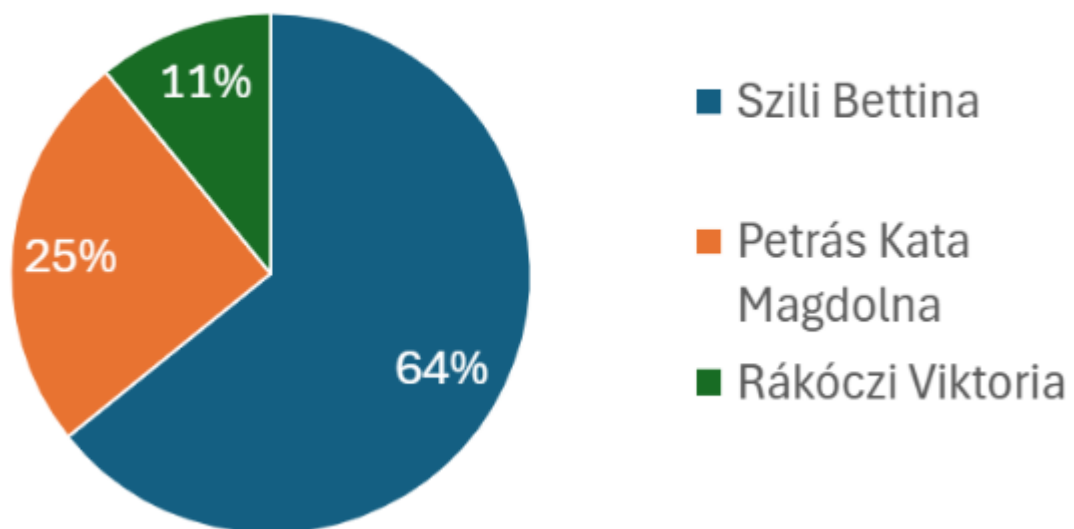
A csapat felosztás pedig a következőképpen nézett ki:

Szili Bettina dolgozott a frontenden, webdesignon, illetve ő volt a csapat project menedzsere,

Petrás Kata Magdolna készítette az adatbázist, illetve az oldalon látható tartalmak is neki köszönhetőek,

Rákóczi Viktória pedig a Backend részért volt felelős.

Órák Eloszlása Csapattagonként



Program működésének a részletes leírása

Program részletes leírása

Ahogy betölt az oldal megjelennek a regisztrációra és bejelentkezésre való opciók. Ha még nincs fiókod, akkor tudsz regisztrálni, és ha sikerül regisztrálnod, akkor be tudsz lépni. Sikeres belépést követően a főoldalra irányít át, ahol található a menüsáv. A menüsávon belül lehet választani, hogy kutyát vagy macskát akarsz örökbe fogadni. Ha nem gépen vagy laptopon vagy, akkor telefonon a hamburger menüt (három horizontális csík a sarokban) lenyitva kiadja, hogy mik a választási lehetőségeid. Amint kiválasztottad milyen állatot akarsz akkor rá tudsz menni a képükre, ahol felugrik egy részletes leírás az állatról. Ha

megtetszik az állat és örökbe akarnád fogadni akkor van egy menhely opció és meg tudod nézni, hogy melyik a legközelebbi menhely hozzád, ellátva elérhetőségekkel, hogy örökbe tudd fogadni. Ha sikeres az adoptáció, akkor haza viheted az állatod, és biztosíthatsz neki egy kényelmesebb életet tele szeretettel és törődéssel.

Tesztelés

Tesztelési fajták

A programozásban két féle tesztelési módszert különítünk el az egyik az automatizált teszt a másik a manuális teszt. Az automatizált teszteket általában olyan funkciók tesztelésére használjuk, amelyeket így egyszerűbben és gyorsabban le tudjuk tesztelni és esetleges bugokat felfedezni, ami eddig nem szűrt szemet. A manuális tesztelés ugyan lassabb, de egyszerűbb elkészíteni.

Jelentőség

Azért nagyon jelentősek a tesztelések, hogy a hibákat még azelőtt észre vegyük, hogy a kész szoftver termékünk üzembe helyezésre kerül, így növeljük annak a minőségét.

Használt technológia

A manuális teszt katalógust Excel táblában készítettük el.

Frontend

A frontendet kizárólag manuálisan teszteltük, amit a Manuális Tesztelés PDF fájlban lehet megtekinteni részletesebben.

Backend

A backendet Postman segítségével teszteltük, aminek eredményeit úgyszintén a Manuális Tesztelés PDF fájlban lehet megtekinteni részletesebben.

Tesztelt endpointok: GET/api/dogs, GET/api/shelters, GET/api/cats



Összegzés és továbbfejlesztési lehetőségek

Tapasztalatok

Nem minden esetben ment jól a kommunikáció, de mindig igyekeztük az ebből adódó vitákat megbeszélni, hogy a munkafolyamatokat ez ne befolyásolja. Rengetegszer elakadtunk, de mindig igyekeztünk segíteni egymásnak. A kezdetek nagyon nehézkesek voltak, de ki szerettük volna hozni a lehető legtöbbet ebből a projectből.

Továbbfejlesztési lehetőségek

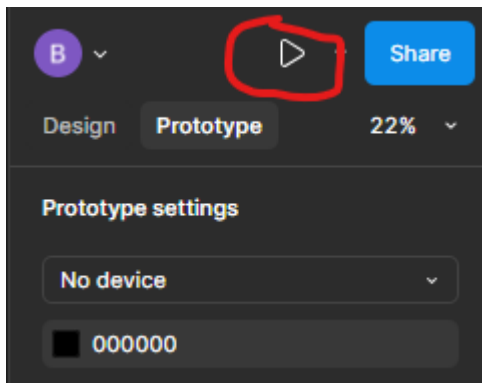
Szeretnénk létrehozni egy olyan algoritmust, ami annyit tesz, hogy miután egy felhasználó regisztrál az oldalunkra, akkor szembe találja majd magát egy kitöltendő teszttel, ami többek között olyan kérdéseket tartalmaz, hogy mit keresnek egy kisállatban, illetve amik segítenek megérteni az ő személyiségüket is egy picit, aztán ennek az eredményének a segítségével ajánljon neki az oldalunk állatokat.

Live-Design

Webdesign elkészítéséhez a Figma nevű programot használtuk, amelyben az oldal pontos kinézetét terveztük meg.

Design terv futtatása prototype modeban:

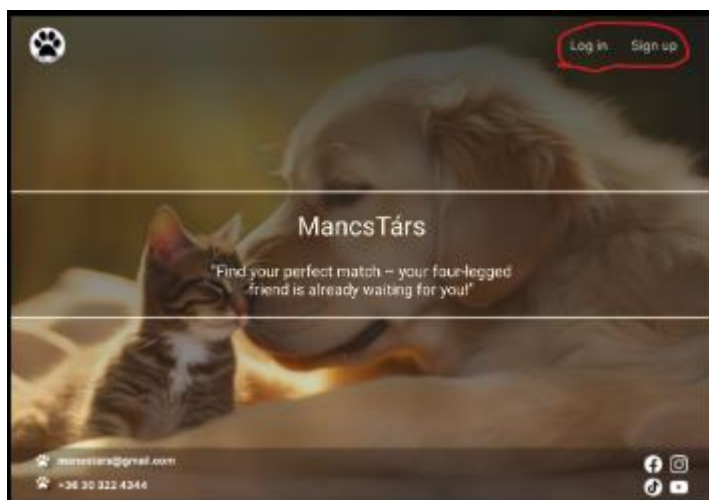
1. Megnyitod az alábbi linket
<https://www.figma.com/design/JbZMkFJKLeqAvhlYasOMM6/MancsT%C3%A1rs?node-id=0-1&t=5vOVC83bFtB09EZp-1>
2. Rá mész a háromszögre jobboldalon fent.



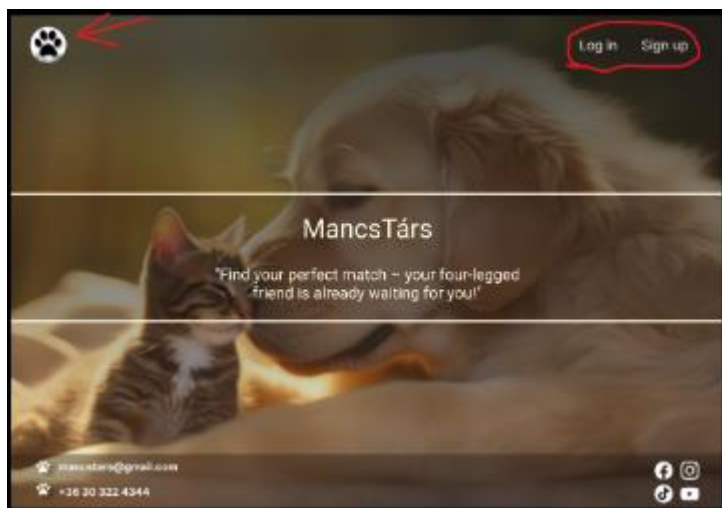
3. Ezzel át dob egy új ablakba, de előtte be kell jelentkezni, mert profil nélkül nem tekinthető meg ez a mód.
4. Ha mind ez megtörtént élénk tárul a kattintható webdesign terv.

Design Funkcionalitása:

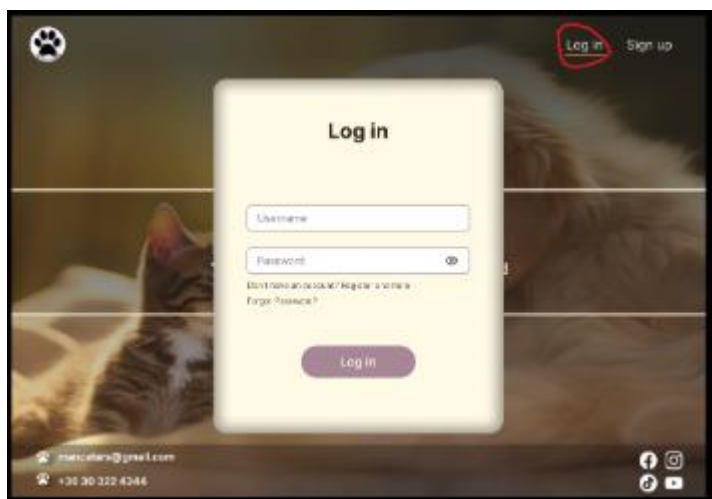
Főoldalon található egy két mezőből álló kattintható navbar az oldal jobboldalán.



A baloldalon pedig a logója az oldalnak.

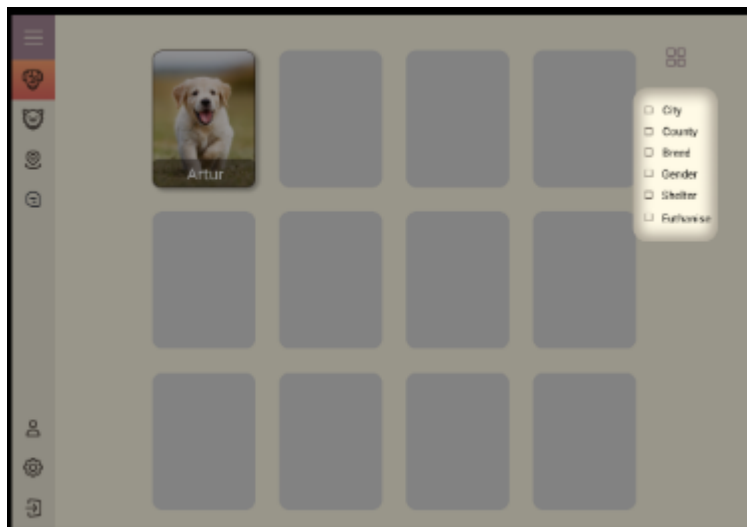


Ha rányomunk a Log in gombra akkor bejön egy popup ablak, ahol be tudsz jelentkezni ugyan így működik a regisztrációs rész is.



A popupban szintén megtalálhatóak azok az opciók, amik minden oldalon is megtalálhatóak.

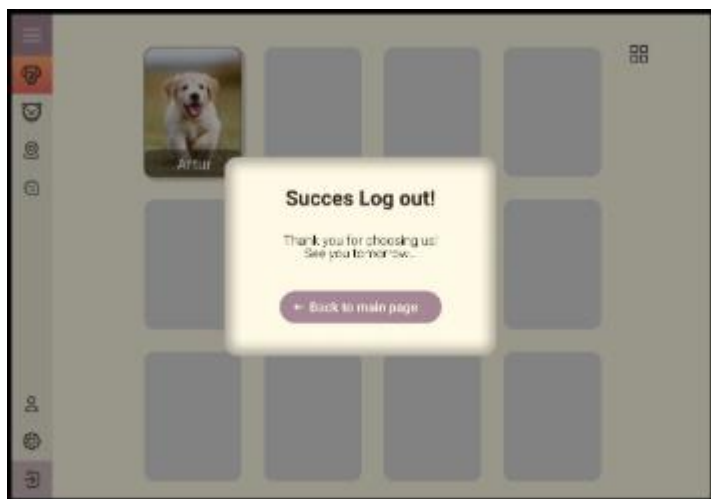
Ha bejelentkezünk akkor bedob minket a kutyás oldalra, ahol található egy sidebar baloldalon, jobboldalon fent egy szűrés opció, középen pedig az állatos kártyák (cicás oldal is így épül fel). Szűrés opciónál megyére, városra, fajtára és nemre lehet szűrni. (szintén ugyanígy van a kutyás oldalon is)



A menhelyes oldalon szintén minden így néz ki csak nincs szűrés opció.



Továbbá van még egy leírás oldal, ahol a fejlesztőkről található pár információ. Van egy profil és egy beállítások opció a szokásos opciókkal, amik egy ilyen fül alatt találhatóak és végezetül van egy kijelentkezés gomb



Telepítési útmutató

Használt programok:

- Visual Studio Code
- Apache Netbeans
- Node.js
- WildFly
- MAMP
- Postman

Használt program nyelvek:

- Java
- MySQL
- HTML
- CSS
- TypeScript

Használt keretrendszerek:

- Angular
- Bootstrap

MAMP letöltése és telepítése:

1. A program letöltése a weboldalukról a számadra legmegfelelőbb operációs rendszerű géphez(macOS/Windows) az alábbi linkről:
<https://www.mamp.info/en/downloads/> .
2. Letöltés után a számítógép fájlkezelőjében található letöltések mappában megnyitjuk a **MAMP-MAMP-PRO-5.0.6.exe** nevezetű fájlt és ezzel elindítjuk a telepítőt. Ha esetleg nem **.exe** fájlt tölt le, hanem **.zip-et** akkor ezt egy **WinRAR vagy egyéb ehhez hasonló program** segítségével ki kell

csomagolni utána fentebb említett módon ugyanúgy el indítható a program telepítése.

3. Fájl kicsomagolása, ha .zip formátumban töltötte le:
 - 3.1. Ha nincs a számítógépen WinRAR vagy egyéb archív fájlokat kezelő program akkor le kell egyet töltened: (WinRAR: <https://www.win-rar.com/start.html?&L=0>).
 - 3.2. Utána ahová letöltötte a fentebb megemlített programot ott a fájlra jobb egérgomb és *Kicsomagolás ide (Extract here)*.
 - 3.3. Ezek után nincs más teendő csak a telepítés elindítása.
4. Ha elindult a telepítő kikell választani, hogy hová szeretnéd telepíteni és utána csak figyelmesen végig kell olvasni mindent és végig kattintgatni.
5. Miután véget ért a telepítés semmi teendő nincs csak el kell indítani a programot.

(Fontos megjegyzések: ha nincs tárhelyed nem fog elindulni a szerver, a fájlokat szigorúan tilos átnevezni, mert nem fog utána működni.)

Backendhez használt programok letöltése és telepítése:

Java Developer Kit letöltése és telepítése:

Ezt az Oracle weboldaláról kell letölteni. Ahhoz, hogy le tud tölteni először regisztrálni kell. Regisztráció során kér cégnevet ide nem kötelező valós cég nevét beírni. Ha megtörtént a regisztráció utána indulhat is a letöltés.

Letöltés menete:

1. Az alábbi linket meg nyitva (<https://www.oracle.com/java/technologies/javase/jdk17-0-13-later-archive-downloads.html>) tölthető le a *17-es jdk* amire szükségünk lesz.
2. Itt lentebb görgetünk és számítógéptől függően azt kell kiválasztani amelyik kompatibilis az operációs rendszerrel (Ha Windows-ot használ akkor azt a fájlt válaszd) ezen felül a *.exe* kiterjesztésű fájlt a legcélszerűbb választani.
3. Ezután ahová letöltöttük azt kiválasztunk és elindítjuk a telepítést.
4. Telepítés befejeztével következhet a Netbeans telepítése.

Netbeans letöltése és telepítése:

A programot a Netbeans oldaláról kell letölteni az alábbi linken megtalálható

<https://netbeans.apache.org/front/main/download/>.

A program letöltése:

1. Az oldalon Download gombra kattintva átdob egy másik oldalra.
2. Ott ki kell választani melyik platformra szeretnéd letölteni (Ha Windows akkor az első).
3. Utána kiválasztod hová szeretnéd ezután indulhat a telepítés.

A program telepítése:

A telepítése az eddigiekhez hasonlóan működik.

1. Megnyitod ahová letöltötted a **.exe** fájlt.
2. Rá nyomsz és elindítod a telepítőt.
3. Figyelmesen elolvassa követed a program utasításait és végig kattintgatod.
4. Telepítés befejeztével már használhatod is a programot.

WildFly letöltése és telepítése:

A program telepítése eltérő a különböző operációs rendszereken ezért mindenhol máshogy kell. A letöltést viszont ugyan arról az oldalról meg tudod tenni az alábbi linken

<https://www.wildfly.org/downloads/>

A program letöltése:

1. A fentebb már említett linkre rá kattintva bedob az oldalukra.
2. Ott lejjebb görgetve ki kell választani a **wildfly 26.1.1.final** részt.
3. Onnan kiválasztod az első **.zip** formátumú fájlt.
4. Ezután kiválasztod, hogy hová szeretnéd és kész is.

A program telepítése Windowsra:

Külön kell engedélyezni az admin felületet is.

1. Csomagold ki a **zip** fájlt egy könyvtárba, pl.: C:\wildfly.
2. Parancssorban navigálj a **bin** mappába, és futtasd a **standalone.bat** fájlt.
3. Hozz létre egy felhasználót ezzel a paranccsal **add-user.bat**
4. Ezután a program végig vezet a felhasználó létrehozásán:
 - **Típus:** válaszd az "a.)" opciót (Management User)
 - **Felhasználónév:** pl. admin
 - **Jelszó:** amit szeretnél
 - A további kérdésekre általában nyomj **Enter-t** vagy válaszd az alapértelmezettet (yes/no)

5. Környezeti változó létrehozása:

- 1.1. Jobb klikk a "**Saját gép**" vagy "**Ez a gép**" ikonra → **Tulajdonságok**
- 1.2. **Speciális rendszerbeállítások**
- 1.3. Kattints a "**Környezeti változók...**" gombra
- 1.4. Alul: **Új** rendszerváltozó Név: WILDFLY_HOME
- 1.5. A **PATH** változót szerkesztve add hozzá: %WILDFLY_HOME%\bin
(vagy a konkrét elérési utat ahová telepítetted)

A program telepítése macOS-re:

1. Csomagold ki az alábbi parancs futtatásával `unzip wildfly-30.0.1.Final.zip -d ~/wildfly`
2. Nyisd meg a terminált, navigálj a `bin` mappába, és futtasd a `standalone.sh` fájlt.
3. Hozz létre egy felhasználót ezzel a paranccsal `add-user.sh`
4. Ezután a program végig vezet a felhasználó létrehozásán:
 - ❖ **Típus:** válaszd az "a.)" opciót (Management User)
 - ❖ **Felhasználónév:** pl. Admin **Jelszó:** amit szeretnél
 - ❖ A további kérdésekre általában nyomj **Enter-t** vagy válaszd az alapértelmezettet (yes/no).

Postman letöltése és telepítése:

A program letöltése:

1. Nyisd meg az alábbi linket <https://www.postman.com/downloads/>
2. Nyomj rá a gombra, ha Windows-os géped van.
3. Válaszd ki hová kívánod letölteni.
4. Ez után indulhat a telepítés.

A program telepítése:

1. Megnyitod a mappát ahová letöltötted.
2. Ki választod a letöltött `.exe` fájlt.
3. Rá kattintasz ezzel elindul a telepítő.
4. Követed a telepítő utasításait és végig kattintgatod.
5. A telepítés befejeztével már használhatod is.

Frontendhez használt programok letöltése és telepítése

Visual Studio Code letöltése és telepítése:

A program letöltése:

1. Menj rá az alábbi linkre <https://code.visualstudio.com/download> .
2. Itt válaszd ki a számodra megfelelő platformot.
3. A gomb megnyomása után válaszd ki hová kívánod letölteni.
4. Ez után jöhet a telepítés.

A program telepítése:

1. Megnyitod azt a mappát ahová letöltötted.
2. Ott kiválasztod a **.exe** fájlt.
3. Rá kattintasz a **.exe** fájlra és ezzel elindítod a telepítőt.
4. Telepítés befejezte után már használhatod is a programot.

Node.js letöltése és telepítése:

A program letöltése:

1. Menj rá az alábbi linkre <https://nodejs.org/en/download>
2. Itt válaszd ki a **Windows installer-t**
3. A gomb megnyomása után válaszd ki hová kívánod letölteni.
4. Ez után jöhet a telepítés.

A program telepítése:

1. Megnyitod azt a mappát ahová letöltötted.
2. Ott kiválasztod a **.exe** fájlt.
3. Rá kattintasz a **.exe** fájlra és ezzel elindítod a telepítőt.
4. Követed a telepítő utasításait és végig kattintgatod.
5. Telepítés befejezte után már nincs más teendő.

A projecthez használt keretrendszerek telepítése:

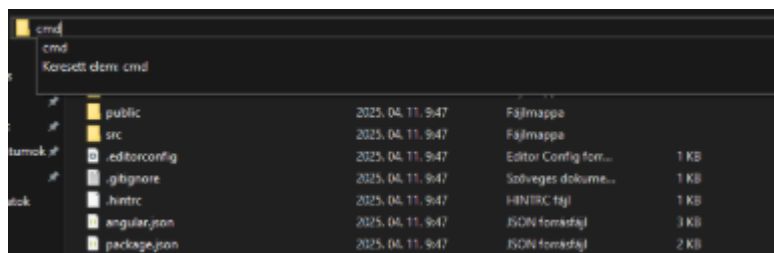
Angular telepítése:

Ehhez már korábban fel kellett telepíteni a node.js-t, mert enélkül nem fog működni

1. Nyisd meg a parancssort rendszergazdaként.
2. Ellenőrizd a node.js verziót `node -v` és `npm -v` parancssal
3. Telepítsd az Angular-t a `npm install -g @angular/cli` parancssal.
4. Ellenőrizd a verziót az `ng version` parancssal.
5. Ezután már használhatod is a keretrendszert.
6. Projectet a `ng new projekt-neve` parancssal hozhatsz létre és ahhoz hogy fejlesztői környezetben futtathasd szükséged lesz a `cd projekt-neve` parancsra és a `code .` Parancsra is.
7. Ez után indulhat is a fejlesztés, ha szervert szeretnél futtatni szükséged lesz az `ng s -o` parancsra is, amit ugyanúgy parancssorban kell lefuttatni azon az elérési útvonalon, ahol a project is található.

Bootstrap telepítése Angularhoz:

1. Nyisd meg a project mappát.
2. Ha bent van a mappában beírod oda, hogy cmd.



3. Ezzel megnyílik a parancssor.
4. Nyisd meg az alábbi linket: <https://getbootstrap.com/> .
5. Ott kimásolod a `npm i bootstrap@5.3.5` parancsot.
6. Ezt a parancsot beilleszted parancssorba.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5608]
(c) Microsoft Corporation. Minden jog fenntartva.

D:\Projekt\MancsTars\manstars>npm i bootstrap@5.3.5
```

7. Miután lefuttattad a parancsot már használhatod is.

Manuális Tesztelés

Id	Task-id	Test-objective	Test-steps	Expected-result	Actual-result	
					Passed	Failed
1	scrum-86	regisztrációs procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázis n belüli "Eljárások" füls megnyitása 3. regisztrációs eljárás lefuttatása	A regisztráció sikeres és hozzá adja a user adatait a táblához	TRUE	
2	scrum-87	addNewCat procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázis n belüli "Eljárások" füls megnyitása 3. addNewCat eljárás lefuttatása	A függvény futása után megjelenik az új macska a táblában	TRUE	
3	scrum-88	addNewDog procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázis n belüli "Eljárások" füls megnyitása 3. addNewDog	A függvény futása után megjelenik az új kutya a táblában	TRUE	

			g eljárás lefuttatása			
4	scrum-89	addNewUse r procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli "Eljárások" fül megnyitása 3. addNewUse r eljárás lefuttatása	A függvény futása után megjelenik az új user a táblában	TRUE	
5	scrum-90	deletedCat procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli "Eljárások" fül megnyitása 3. deletedCat eljárás lefuttatása	A függvény futása után a kiválasztott macska adatai törölődnek a táblából	TRUE	
6	scrum-91	deletedDog procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli "Eljárások" fül megnyitása 3. deletedDog eljárás lefuttatása	A függvény futása után a kiválasztott kutya adatai törölődnek a táblából	TRUE	
7	scrum-92	deletedUse r procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli	A függvény futása után a kiválasztott user adatai törölődnek a táblából	TRUE	

			"Eljárások" fül megnyitása 3. deletedUser eljárás lefuttatása			
8	scrum-93	getAllCat procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázison belüli "Eljárások" fül megnyitása 3. getAllCat eljárás lefuttatása	A függvény futása után kilistázza a cats tábla adatait	TRUE	
9	scrum-94	getAllDog procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázison belüli "Eljárások" fül megnyitása 3. getAllDog eljárás lefuttatása	A függvény futása után kilistázza a dogs tábla adatait	TRUE	
10	scrum-95	getAllShelter procedure tesztelése	1. phpmyadmin megnyitása 2. Adatbázison belüli "Eljárások" fül megnyitása 3. getAllShelter eljárás lefuttatása	A függvény futása után kilistázza a shelters tábla adatait	TRUE	
11	scrum-96	getAllUser procedure tesztelése	1. phpmyadmin	A függvény futása után	TRUE	

			megnyitása 2. Adatbázis n belüli "Eljárások" fül megnyitása 3. getAllUser eljárás lefuttatása	kilistázza a users tábla adatait		
12	scrum-97	getNewPas sword procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbázis n belüli "Eljárások" fül megnyitása 3. getNewPas sword eljárás lefuttatása	A függvény futása után megváltozik a kiválasztott user jelszava	TRUE	
13	scrum-98	login procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbázis n belüli "Eljárások" fül megnyitása 3. login eljárás lefuttatása	A függvény futása után nem dob error-t a phpmyadmi n	TRUE	
14	scrum-99	updateCat procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbázis n belüli "Eljárások" fül megnyitása 3. updateCat	A függvény futása után frissülnek a kiválasztott macska adatai	TRUE	

			eljárás lefuttatása			
15	scrum-100	updateDog procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli "Eljárások" fül megnyitása 3. updateDog eljárás lefuttatása	A függvény futása után frissülnek a kiválasztott kutya adatai	TRUE	
16	scrum-101	updateUser procedure tesztelése	1. phpmyadmi n megnyitása 2. Adatbáziso n belüli "Eljárások" fül megnyitása 3. updateUser eljárás lefuttatása	A függvény futása után frissülnek a kiválasztott user adatai	TRUE	
17	scrum-77	login függvények tesztelése	1. Postman megnyitása 2. "POST" kérésbe beírjuk az alábbi linket: http://127.0.0.1:8080/Mancs_Tars-1.0-SNAP/webresources/user/login 3. Megadjuk a Body-n belül raw (JSON) formátumb an a	Lekérés után 200-as HTTP-kódot kapunk vissza	FALSE	Model-réteg nem éri el az adatbázisból érkező adatokat, így az EntityManager sem tudja listává alakítani azokat

			megadott user email címét és jelszavát 4. API-kérés küldése			
18	scrum-78	login tesztelése frontenden	1. Projekt megnyitása böngészőben 2. "Login" gomb megnyomása 3. Bejelentkezés a kiválasztott user adataival	A "Login" gomb megnyomása után átirányít minket a főoldal "dogs" almenüjére	TRUE	
19	scrum-79	regisztrációs függvények tesztelése	1. Postman megnyitása 2. "POST" kérésbe beírjuk az alábbi linket: http://127.0.0.1:8080/Mancs_Tars-1.0-SNAP/webresources/user/register 3. Megadjuk a Body-n belül raw (JSON) formátumban a megadott user felhasználó nevét, email címét és jelszavát 4. API-kérés küldése	Lekérés után 200-as HTTP-kódot kapunk vissza	FALSE	Model-réteg nem éri el az adatbázisból érkező adatokat, így az EntityManager sem tudja listává alakítani azokat

20	scrum-102	regiszter tesztelése frontenden	1. Projekt megnyitása böngészőbe n 2. "Register" gomb megnyomás a 3. Új fiók létrehozása	A "Register" gomb megnyomás a után átirányít minket a főoldal "dogs" almenüjére	TRUE	
21	scrum-80	getAllDog api tesztelése	1. Projekt megnyitása böngészőbe n 2. Bejelentkez és egy tetszőleges fiókkal 3. "dogs" almenüre érkezés	A kutyák adatai megjelenne k az oldalon	TRUE	
22	scrum-81	getAllCat api tesztelése	1. Projekt megnyitása böngészőbe n 2. Bejelentkez és egy tetszőleges fiókkal 3. "cats" almenü megnyitása	A macskák adatai megjelenne k az oldalon	TRUE	
23	scrum-82	getAllShelte rs api tesztelése	1. Projekt megnyitása böngészőbe n 2. Bejelentkez és egy tetszőleges fiókkal 3. "shelters" almenü megnyitása	A menhelyek adatai megjelenne k az oldalon	TRUE	
24	scrum-83	getAllUser api tesztelése	1. Postman megnyitása 2. "GET"	Lekérés után 200-as	FALSE	Model-réteg nem éri el az

			kérésbe beírjuk az alábbi linket: http://127.0.0.1:8080/Mancs_Tars-1.0-SNAP/webresources/user/getAllUsers 3. API- kérés küldése	HTTP-kódot és az egész users tábla tartalmát kapjuk vissza		adatbázisból érkező adatokat, így az EntityManager sem tudja listává alakítani azokat
25		login gomb tesztelése	1. Projekt megnyitása böngészőbe n 2. "Login" gomb megnyomás a	Átírányít minket a Login formhoz	TRUE	
26		hoverek tesztelése	1. Projekt megnyitása böngészőbe n 2. Gombok és navbar föle visszük az egeret	A színük megváltozik és a main page-en a Login és Register gombok alatt aláhúzás jelenik meg	TRUE	
27		authguard tesztelése üres mezőkkel	1. Projekt megnyitása böngészőbe n 2. A Login formon lévő mezőket üresen hagyjuk 3. "Login" gomb megnyomás a	A gomb megnyomás a után globális error-t kapunk	TRUE	
28		reszponzivitás	1. Projekt megnyitása böngészőbe n 2. Jobb	Állítgatás közben az oldalon lévő elemek átméretező	FALSE	A "Shelters" oldal elemei nem változnak, a többi

			gombbal rákattintunk az oldalra 3. "Vizsgálat" opció kiválasztása a felugró menüből 4. Az oldalt lévő csúszka segítségével átállítjuk az oldal méretét	dnek, a navbarból szendvics menü lesz, a képek mérete és layoutja megváltozik		viszont reszponzív
29		authguard tesztelése invalid adatokkal	1. Projekt megnyitása böngészőbe n 2. A Login formon lévő mezőkbe nem valid adatokat adunk meg (nem létező email, rossz jelszó) 3. "Login" gomb megnyomás a	A gomb megnyomás a után globális error-t kapunk	TRUE	
30		authguard tesztelése valid adatokkal	1. Projekt megnyitása böngészőbe n 2. A Login formon lévő mezőkbe valid adatokat adunk meg 3. "Login" gomb megnyomás a	A gomb megnyomás a után átírányít minket a főoldal "dogs" almenüjére	TRUE	