

# Szakvizsga-dolgozat

## Sertések adatbázisba való vezetése

Készítette: Kantor Rolland-Robert  
Felkészítő tanár: Jakab Irma Tünde  
Iskola: Bolyai Farkas Elméleti Líceum  
Osztály: XII.G

2021

# Tartalomjegyzék

## Tartalom

1.Témaválasztás .....	3
2.Rendszer követelmények.....	3
A. Hardver igény .....	3
B.Szoftver igény.....	3
3.Felhasználói kézikönyv.....	4
1.Sertés .....	7
.....	9
2.Megtermékenyítés .....	9
3.Fialás .....	10
4.Takarmány.....	11
5.Szűrés.....	12
Adatbázis gombok .....	12
Aktuális adatok.....	13
4.Programozói kézikönyv.....	14
5.Fejlesztési lehetőségek .....	17
6.Szakirodalom.....	17

## 1.Témaválasztás

A dolgozatom témája a sertések nyilvántartása. Azért választottam ezt a témát, mert egy megoldást szerettem volna létrehozni egy személyes problémára, ami a sertések nyilvántartása. A családomban pár éve gondot okoz a sertések nyilvántartása, ugyebár évente csak pár alkalommal kell vele foglalkozni, de ilyenkor mindig több darab mappát és több tucat papírt kell átvizsgálni, hogy megfelelően tudjuk vezetni a sertéseket, viszont így is történnek néha hibák. Emellett a papírmunka időigényes is. Ezzel a weboldallal azt szerettem volna elérni, hogy a felhasználók gyorsan és könnyedén tudják nyilvántartani a sertéseiket, emellett, hogy az adatok digitális formában legyenek meg egy felületen.

## 2.Rendszer követelmények

### A. Hardver igény

Ajánlott asztali számítógépet vagy laptopot használni. A biztonság kedvéért kezdetnek elegendő pár GB tárhely, ez felhasználástól függően ( sok adat bevitel során ) lehet majd bővülésre szórul.

### B.Szoftver igény

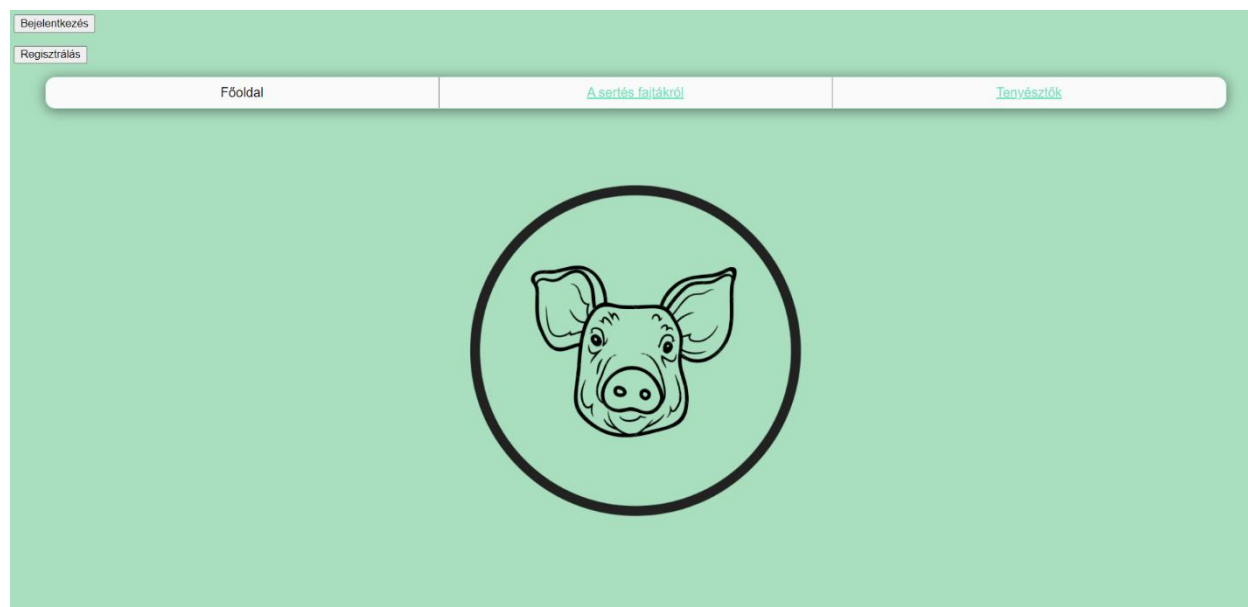
A program megírására a következő programozási nyelveket használtam:

- HTML – kódnyelvvél valósítottam meg a weboldal felületi elemeit, a weboldalak szerkezetét, illetve az oldalak szöveges tartalmának a feltöltése, emellett a menüsor és a különböző oldalak tartalmainak meghatározása itt olvasható
- PHP – segítségével történik a műveletek elvégzése, összeköthetjük az oldal felületét és fizikai részét, itt történik a bevitt adatok rögzítése, a feltételeknek való megfelelés ellenőrzése, valamint az utasítások futtatása, a lekérdezések végrehajtása is
- SQL – az adatbázissal való műveletekre használtam, adatfeltöltés, lekérdezés stb.
- JAVASCRIPT ( JS ) - a weboldal dinamikussá tételéért használtam, mivel ez a programozási nyelv állandóan fut a háttérben, amikor meg van nyitva az oldal
- CSS – ez a programozási nyelv kapcsolódik hozzá a HTML-hez, a CSS egy stíluslap, ami leírja egy weboldal megjelenését. A CSS segítségével, formáltam meg a HTML elemek megjelenését, például táblázat és gomb kinézete, bekezdések, szövegek, képek elhelyezése

A program működtetéséhez a MySql és Apache szerverekre van szükség, amiket a XAMPP szoftveren keresztül érhetőek el. Az oldalt a localhost-on keresztül kell megnyitni.

### 3.Felhasználói kézikönyv

A felhasználó a weboldalt megnyitva a fő oldalra kerül (lásd 3.1 ábra),



3.1 ábra Főoldal

ahol az oldal két funkcióját tudja használni: a sertés fajok megtekintését, ahol röviden be vannak mutatva a fajok (lásd 3.2 ábra),

<b>Báznai</b>
<p>A báznai sertésfajtát a mangalica koca és a berk kanok keresztezésével tenyésztették ki 1872-től, az erdélyi Báznán (ma Szeben megye).</p> <p>A mangalicánál jobb hőstermelésre képes fajtát hamar megszerették, főként Medgyes, Segesvár, Fogaras környékén lett a gazdaságok kedvence. Erdély elcsatolásával az állomány Romániába került, Magyarországon számottevő létszáma nincs.</p> <p>Megjelenése érdekes, a teste nagyrészt, a test elejét és a fari részét fekete szőr borítja, míg a test közepe fehér, ezért "öves disznónak" is nevezik.</p> <p>A báznai sertés szaporasága közepes, általában 7-11 öves malacot ellik. A produktív hosszú élettartam a kocáknál figyelemre méltó: 8-12 ellés/ élet.</p>
<b>Duroc</b>
<p>Gyors növekedési erélyű, nagy vágási súlyra hizlalható, alacsony fajlagos takarmányértékesítéssel bíró, tömeges hússertés.</p> <p>Húsformái kifejezetten karaja terjedelmes, sonkája széles és mélyre húzódó, lapockája telt.</p> <p>Erős szervezeti szilárdságú, teste és csontozata robusztus, rámás, háta feszes, a lábai erőteljesek.</p> <p>A nyak rövid, a mellkas hengeres. Feje jellegzetesen kúpos alakú, kissé rövid, széles homlokkal. Az orrhat enyhén homorú. Fülei rövidek, a haránt középvonaltól megtörtek, előre billenők.</p> <p>Bőre rózsaszín, körmei palaszürkék, túrókarimája sötétbarnától a kékes színűig terjedő árnyalatú, pigmentált. Szőre sima és sűrű, színe a sötét cseresznyepestőtől a sárgás világos vörösig terjed.</p>
<b>Pietrain</b>
<p>Kiemelkedően jó vágóértékű, alacsony fehérarú %-ú, „szuperizmolto sertés”. A karaj, a combok és a lapockák extrém izmoltságúak.</p> <p>A hát és a far „barázdált”. A bőr és a szalonnaköpeny vékony, az izmok jól látszanak a combok felületén, az erek kidudorodnak. A csontozat finom, szilárd, a hátvonal feszesen egyenes. A fej könnyű, rövid, esetleg középhosszú, a profilvonal enyhén tört, a homlok széles. A fülek rövidek, felállóak, de lehetnek előre hajlóak is.</p> <p>Szabálytalan tarka sertés, szürke, vagy fehér testfelületen pigmentált foltokkal. A szőr színe a fehértől a feketéig folytonos változó, finom, sima lefutású, néha nagyon</p>

3.2 ábra Sertés fajták megtekintése

illetve a tenyésztők funkciót, ahol megtudja tekinteni az addig regisztrált sertéstenyésztők/-tartók elérhetőségeit (lásd 3.3 ábra).

Vezetőknév: <b>Kantor</b> Keresztnév: <b>Rolland</b> Farm neve: <b>MGL</b> Telefonszám: <b>0748576803</b> Megye: <b>Maros</b> Település: <b>Sáromberke</b> Cím: <b>D15</b>	Vezetőknév: <b>Szabo</b> Keresztnév: <b>Botond</b> Farm neve: <b>Botex Farm</b> Telefonszám: <b>0748556868</b> Megye: <b>Hargita</b> Település: <b>Korond</b> Cím: <b>Fout 115</b>	Vezetőknév: <b>Kovács</b> Keresztnév: <b>János</b> Farm neve: <b>Kovács farm</b> Telefonszám: <b>074886503</b> Megye: <b>Maros</b> Település: <b>Marosvásárhely</b> Cím: <b>Kossuth utca 11</b>
Vezetőknév: <b>Váradi</b> Keresztnév: <b>Domokos</b> Farm neve: <b>Malomház</b> Telefonszám: <b>0744232367</b> Megye: <b>Maros</b> Település: <b>Fickó</b> Cím: <b>Főút 205</b>		

3.3 ábra Tenyésztők

A két alapfunkció mellett még van két gomb: a Regisztráció és a Bejelentkezés.

Fontos, hogy aki még nem regisztrált az tegye meg, mert a bejelentkezés csak azután történhet meg.

A bejelentkezéshez az email-cím szükséges és a megadott jelszó (lásd 3.4 ábra).

3.4 ábra Bejelentkezés

# Regisztráció

Email-cím

Jelszó

Sorszám

Vezetéknév

Keresztnév

Farm neve

Telefonszám

Megye

Település

Lakcím/farm címe

3.5 ábra Regisztráció

A regisztrálás egyszerűen történik, minden mező mellett írja, hogy milyen adatot kell beírni az adott mezőbe (lásd 3.5 ábra). A felhasználó ha rossz formátumú e-mail címet vagy telefonszámot ad meg a weboldal jelzi majd. A sorszámot a weboldal adja meg. Az összes mező kitöltése kötelező, ha kitöltöttük a mezőket a Regisztrálás gombbal véglegesítjük az adatokat és regisztrálunk. Ha nem töltöttük ki az összes mezőt, akkor hibaüzenetet fog adni az üres mezőnél. Ha a felhasználó regisztrál, akkor úgy tekintjük, hogy tenyésztő, illetve tartó és adatbázist hozunk létre neki, ami az oldal további funkcióihoz kapcsolódik.

Ha a felhasználó már bejelentkezett több funkció érhető el számára. Amikor bejelentkezünk a felhasználói oldalra továbbít az oldal (lásd 3.6 ábra). Ezen az oldalon megjelennek azok az elemek, amik a szimpla felhasználónál ( aki nincs bejelentkezve ) is

elérhetőek, viszont eltűnt a Bejelentkezés, illetve a Regisztrálás gomb és helyettük már csak egy Kijelentkezés gomb található. Ez a menüsor alatt megtalálhatóak a Sertés, Fialás, Megtermékenyítés és Takarmány funkciók, ezekkel a funkciókkal tudjuk megvalósítani az állatok nyilvántartását, az adatokat bevezetését az adatbázisba.

A Sertés, Fialás és Megtermékenyítés funkciók elengedhetetlenek az állomány vezetésénél. A Takarmány funkciót plusz opció a gazdáknak, hogy napirendben legyenek a takarmányuk mennyiségével.



3.6 ábra Bejelentkezett felhasználó főoldal

## 1.Sertés

A Sertés funkció segítségével történik meg a sertések nyilvántartása egyik adattáblába. Egy egyszerű űrlap, ahol minden mezőt kötelező kitölteni (lásd 3.7 ábra). Az első elem a sorszám ami azt jelöli, hogy hányadik sertés a táblában, ezzel a felhasználó nem kell foglalkozzon, az oldal mindig a megfelelő sorszámot adja meg.

A következő három mező a sertés azonosító és szülei azonosítója, amik kötelezően 2 nagy betűből és 12 darab számjegyből kell álljanak. Mivel ezt a weboldalt a romániai magyar gazdák fogják használni és szinte az összes esetben Romániában bejegyzett sertésekkel foglalkoznak, ezért az azonosító szigorúan RO-val kezdődik. A placeholderben a felhasználó láthatja az azonosító formátumát.

A bemeneti adat formátumát reguláris kifejezésekkel vizsgálja meg az oldal, ha véglegesítjük és nem jó az azonosító akkor hibaüzenetet ír ki (lásd 3.8 ábra). Emellett a

weboldal ellenőrzi, hogy különbözik-e a három sertés azonosító, ha nem különböznek, akkor az oldal figyelmezteti a felhasználót (lásd 3.9 ábra).

A következő mező a sertés születési dátuma, amit a naptárból kell kiválasztani, a legfrissebb dátum mindig az aktuális nap. Ezután meg kell adni, hogy milyen fajta a sertés, ezt a megadott opciók közül kell kiválasztani. Ezt követően meg kell adni a sertés nemét, illetve a típusát ( Malac, Süldő, Hízó, Kan, Koca).

A weboldal figyel arra, hogy ha a felhasználó a nemnek a Hímet választotta és típusnak a Kocát, akkor az oldal átállítja a nemet Nősténynek, ugyanezt lekezeli, ha a nem Nőstény és a típus Kan, ekkor a nemet átállítja Hímmé. Végül a Véglegesít gombbal adjuk hozzá az adatokat.

A sertés törlése úgy történik, hogy az állományban levő sertések azonosítói közül ki kell választani a törölni kívánt azonosítót, majd a Törlés gombbal véghezvinni.

## Sertés hozzáadása

Sorszám:

Sertés azonosítószáma:

Anya azonosítószáma:

Apa azonosítószáma:

Születési dátum:  

Nem: ☒ Hím ☐ Nőstény

Fajta:  

Típus:  

## Sertés törlése

Azonosítószám  

3.7 ábra Sertések funkció



Sertés azonosítószáma:

Anyá azonosító  Adja meg a helyes azonosítószámot!

3.8 ábra Hibaüzenet

localhost says

Az azonosítók szigorúan különbözőek kell legyenek!  
Kérjük vizsgálja meg őket, mert legalább két azonosító nem különbözik egymástól!

**Sertés hozzáadása**

Sorszám:

Sertés azonosítószáma:

Anyá azonosítószáma:

Apa azonosítószáma:

3.9 ábra Figyelmeztetés

## 2.Megtermékenyítés

# Megtermékenyítés hozzáadása

Sorszám:

Koca azonosítószáma:


Kan azonosítószáma:

Időpont:  

3.10 ábra Megtermékenyítés funkció

A megtermékenyítésnél a koca és a kan azonosítóját és a megtermékenyítés időpontját kell megadni. Az azonosítóknál ugyancsak kell figyelni a helyességre, ugyanúgy működik, mint az előző funkciónál. A sorszámot a weboldal adja meg. Végül a Véglegesít gombbal adjuk hozzá az adatokat.

### 3.Fialás



**Fialás hozzáadása**

Fialás sorszáma: 4

Anya azonosítószáma: pl:RO3568014659

Apa azonosítószáma: pl:RO3568014659

Megtermékenyítés sorszáma: 0

Időpont: 05/03/2021

Fialt malacok száma: 0

Felnevelt malacok száma: 0

Fiú malacok száma: 0

Lány malacok száma: 0

Elválasztási időpont: 05/03/2021

Véglegesít

3.11 ábra Fialás funkció

A Fialás funkciónál pár adatok kell megadni a fialásról (lásd 3.11 ábra). A megszokott anya (koca) és apa (kan) azonosítókat kell megadni, illetve a két állatnak a megfelelő megtermékenyítésének a sorszámát, amit a felhasználó a megtermékenyítés táblában tud megnézni. Fontos, hogy a megfelelőt adja meg, mivel ugyanannak a két állatnak még lehetett közös megtermékenyítése.

Majd meg kell adni a fialás időpontját.

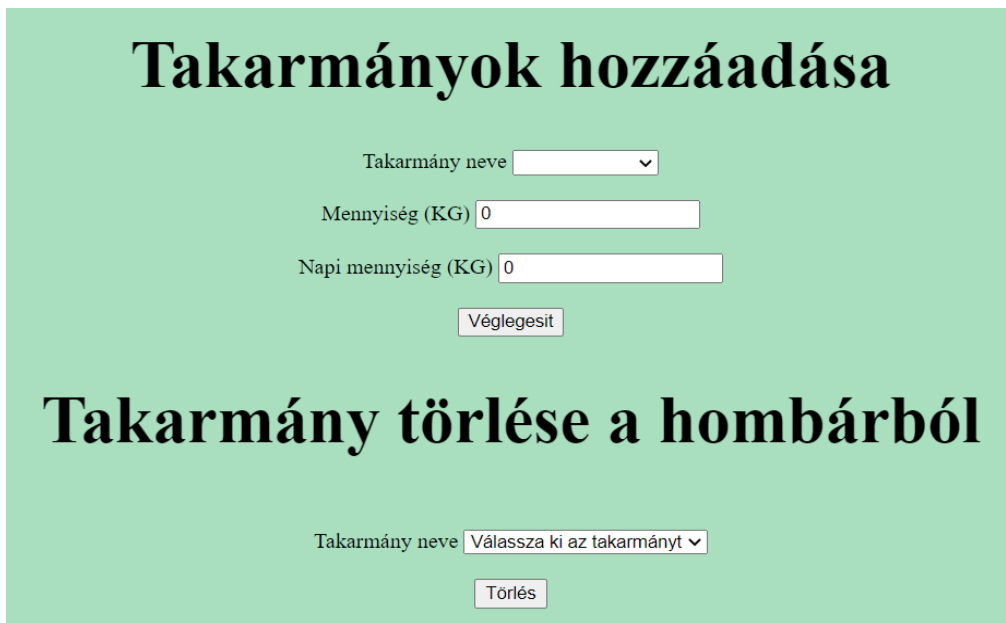
Ezt követően meg kell adni azt, hogy hány malac született, ebből hány élő malac, akiket felnevelt a koca, majd azt kell megadni, hogy a felnevelt malacok, hogy oszlanak el

nemileg (fiú-lány vagy hím-nőstény). Itt a felhasználó figyelmébe ajánljuk, hogy kell figyelni a matematikára, de ebben segítséget nyújt a weboldal is.

Az utolsó mezőben azt kell megadni, hogy a malacokat mikor választotta el a gazda a kocától. Végül a Véglegesít gombbal adjuk hozzá az adatokat.

Ajánlott az adatokat feljegyezni és a fialás után pár héttel feltölteni az adatokat a weboldalra.

#### 4.Takarmány



**Takarmányok hozzáadása**

Takarmány neve

Mennyiség (KG)

Napi mennyiség (KG)

Véglegesít

**Takarmány törlése a hombárból**

Takarmány neve

Törlés

3.12 ábra Takarmány

A takarmány hozzáadása funkciónál ki kell választani a lehetőségek közül a kívánt takarmányfajtát, mivel rengeteg takarmányt használnak manapság a gazdák, ezért csak az legelterjedtebbek szerepelnek a lehetőségek. Ezt követően meg kell adni az összmennyiséget és a napi mennyiséget amit elfogyasztanak az állatok. Ha már szerepel az adott takarmány akkor a mennyiséget hozzáadja az eddigi mennyiséghez a weboldal. Végül a Véglegesít gombbal adjuk hozzá az adatokat.

A takarmány törlésénél pedig ki kell választani az eddig hozzáadott takarmányok közül amelyiket törölni szeretné. Végül a Törlés gombbal törli az adott takarmányt.

## 5.Szűrés

A felhasználó bal oldalt ki kell válassza milyen fajtát, típust, illetve nemet szeretne szűrni, ezután pedig a rendezési szempontot (lásd 3.13 ábra). A weboldal a rendezési szempontokat a típus szerint teszi elérhetővé, illetve nem elérhetővé. Miután megnyomja Szűrés gombot az eredmény jobb oldalt fog megjeleníteni.

**Szűrő**

Fajta kiválasztása  
Vörös mangalica ▼

Típus kiválasztása  
Malac ▼

Nem kiválasztása  
Mindkettő ▼

Rendezési szempont  
Életkor szerint csökkenő ▼

Malacozási átlag  
(az értéknél nagyobb)  
Írja be a malacozási átlagot

Fialások száma  
(az értéknél nagyobb)  
Írja be a fialások számát

Szűrés

SORSZÁM	SERTÉS AZONOSÍTÓ	ANYA AZONOSÍTÓ	APA AZONOSÍTÓ	SZÜLETÉSI DÁTUM	FAJTA	NEM	TÍPUS
6	RO3567632300	RO3568048864	RO3568014659	2020-03-25	Vörös mangalica	Nőstény	Malac
5	RO3567632203	RO3568048863	RO3568014659	2019-12-20	Vörös mangalica	Nőstény	Malac

3.13 ábra Szűrő

## Adatbázis gombok

- A. Sertések adatbázis
- B. Fialás adatbázis
- C. Megtermékenyítés adatbázis
- D. Takarmány adatbázis

Adatbázisok

SERTÉS ADATBÁZIS FIALÁS ADATBÁZIS MEGTERMÉKENYÍTÉS ADATBÁZIS TAKARMÁNY ADATBÁZIS

3.14 ábra Adatbázis gombok

A gombokat megnyomva megnyílik az adott adatbázis egy új ablakban (lásd 3.15 ábra). Azért új ablakban nyílik meg, hogy a felhasználó adatfeltöltés közben könnyen meg tudjon nézni egy szükséges adatot az adatbázisban például: a megtermékenyítés sorszámát a Fialás funkcióban.

SORSZÁM	AZONOSÍTÓ	ANYA AZONOSÍTÓ	APA AZONOSÍTÓ	SZÜLETÉSI DÁTUM	FAJTA	NEM	TÍPUS
1	RO3568014659	RO3558134434	RO3558682866	2018-03-02	Vörös mangalica	Hím	Kan
2	RO3568048863	RO3567601942	RO3570285758	2018-04-27	Vörös mangalica	Nőstény	Koca
3	RO3568048864	RO3567601942	RO3570285258	2018-04-27	Vörös mangalica	Nőstény	Koca
4	RO3568048865	RO3567601942	RO3570285758	2018-04-27	Vörös mangalica	Nőstény	Koca
5	RO3567632203	RO3568048863	RO3568014659	2019-12-20	Vörös mangalica	Nőstény	Malac
6	RO3567632300	RO3568048864	RO3568014659	2020-03-25	Vörös mangalica	Nőstény	Malac

3.15 ábra Sertés adatbázis

## Aktuális adatok

Az utolsó elem, ami látható a főoldalon az az Aktuális adatok táblázat, ahol a felhasználó számadatokat kap meg az állományáról.

Akinek üres az adatbázisa:

SERTÉSÁLLOMÁNY	KOCÁK SZÁMA	HÍZOK SZÁMA	FIATAL EGYEDEK SZÁMA	FAJTÁK SZÁMA
0	0	0	0	0

3.16 ábra Aktuális adatok - üres adatbázis

Akinek nem üres az adatbázisa:

SERTÉSÁLLOMÁNY	KOCÁK SZÁMA	HÍZOK SZÁMA	FIATAL EGYEDEK SZÁMA	FAJTÁK SZÁMA
6	3	0	2	1

3.17 ábra Aktuális adatok - nem üres adatbázis

## 4. Programozói kézikönyv

Ahogy már említettem a weboldal megírásához 5 programozási nyelvet használtam: HTML, PHP, SQL, JAVASCRIPT és CSS.

Az weboldal PHPMyAdmin adatbázis-kezelő szerveret használja, az összes funkcióhoz kapcsolódik az adatbázishoz. Az adatbázisban alaphoz két adattábla van, egy takarmánytípus néven, ahol a takarmány típusok találhatóak azonosítóval és névvel ellátva, és egy felhasználó néven, ami üres. Ez regisztráció során fog feltöltődni, ide kerülnek a bejegyzett felhasználó adatai. Regisztráció során létrehozunk a felhasználónak négy táblát, amelyeknek nevei végén szerepel a felhasználó sorszáma, így tudjuk megkülönböztetni őket. Például az első regisztrálnak a következő adattáblái lesznek: `sertesek_1`, `fialas_1`, `megtermekenyites_1`, `takarmany_1`.

```
$sorszam=$_POST['sorszam'];
$id=$sorszam;
$tablanev="fialasok_".$id;
$tabla=mysqli_query( $conn, " CREATE table $tablanev
(
  Sorszam int(255) not null,
  Koca_azonosito varchar(13) not null,
  Kan_azonosito varchar(13) not null,
  Megtermekenyites_sorszam int(255) not null,
  Idopont date not null,
  Fialt_malacok int(255) not null,
  Felnevelt_malacok int(255) not null,
  Fiu int(255) not null,
  Lany int(255) not null,
  Elvalasztasi_idopont date not null
); " );
$tabla=mysqli_query( $conn, " ALTER TABLE $tablanev CONVERT TO CHARACTER SET utf8 COLLATE utf8_general_ci " );
```

4.1 ábra Fialás adattábla létrehozása

A bejelentkezés során egy `$_SESSION` tömböt használok, amikor a felhasználó bejelentkezett akkor a tömb felhasználó eleme értéket kap: „`$_SESSION['userid']=$id_akt`”, jelen esetben a felhasználó sorszámát. Ez azért szükséges, hogy a weboldal tudja, hogy melyik adatbázissal kell dolgozzon, illetve még azért mert a nem bejelentkezett felhasználóknak kevesebb opció elérhető és ezt is a „`$_SESSION['userid']`”-val valósítja meg (lásd 4.2 ábra).

```

<?php
    session_start();
    if($_SESSION['userid'] == 0){
        ?>
        <?php
            ?>
            <td>
                <form action="bejelentkezes.php" method="get">
                    <button class="btn2" type="submit">Bejelentkezés</button>
                </form>
                <form action="regisztracio.php" method="get">
                    <button class="btn2" type="submit">Regisztrálás</button>
                </form>
            </td>
        <?php
            }
        else {
            ?>
            <form action="kijelentkezes.php" method="get">
                <button class="btn2" type="submit"> Kijelentkezés </button>
            </form>
        <?php
            }
        ?>
    }
}

```

4.2 ábra A Bejelentkezés, a Regisztrálás és a Kijelentkezés gombok megjelenítése

JavaScriptet használók az űrlapok mezőinél a helyes bevétel érdekében. Van olyan eset amikor JS-el kérem le az azonosítók értékét és ha nem különböznek, akkor figyelmeztetem a felhasználót (lásd 4.3 ábra).

```

//1. ha az azonosito mezo változik

//lekerem az azonositot
var azonosito = document.querySelector(".sertesazon1") //class nev

// ezekkel mondtam mit csináljon mikor egy adott esemény van
azonosito.addEventListener('change', azonosito_1_esemeny_kezelo)

function azonosito_1_esemeny_kezelo(){

    // Ezzel lekerem a mezonek az erteket
    var anya_azonosito = document.getElementById("sertesazon2") //id
    var apa_azonosito = document.getElementById("sertesazon3") //id
    var semmi="";
    var ok=0;

    if( anya_azonosito.value!=semmi)
        if (apa_azonosito.value!=semmi)
        {
            if( (azonosito.value != anya_azonosito.value) && (azonosito.value != apa_azonosito.value) && (anya_azonosito.value != apa_azonosito.value) )
            {
                ok=1;
            }
            else
            {
                ok=0;
            }
        }

        if (ok==0) alert("Az azonosítók szigorúan különbözőek kell legyenek! \nKérjük vizsgálja meg őket, mert legalább két azonosító nem különbözik egymástól!")
    }
}

```

4.3 ábra Azonosítók leellenőrzése a Sertesek funkciónál

JavaScripttel oldom meg a Sertések funkcionál a típushoz a megfelelő nem választását.

```
//ha a típus kan/koca
var tipus = document.querySelector(".tipus") //class nev

tipus.addEventListener('change', tipus_ertek_atallitas_esemeny_kezelo)

console.log(tipus.value);
function tipus_ertek_atallitas_esemeny_kezelo(){

    if( tipus.value=="Kan" )
    {
        var sex = document.getElementById("sex") //id
        document.getElementById("him").checked = true;
        document.getElementById("nosteny").checked = false;
        //sex.setAttribute("value", "Him" )
    }
    if( tipus.value=="Koca" )
    {
        var sex = document.getElementById("sex") //id
        document.getElementById("him").checked = false;
        document.getElementById("nosteny").checked = true;
    }
}
```

4.4 ábra Típus-Nem

A Szűrő funkcionál is JavaScriptet használok a mezők aktívvá/inaktívvá tevéséért, ami típus szerint változik (lásd 4.5 ábra).

```
//lekerem a típus mezo tartalmat
var tipus = document.querySelector(".tipus") //class nev

// ezekkel mondtam mit csináljon mikor egy adott esemeny van
tipus.addEventListener('change', tipus_esemeny_kezelo)

function tipus_esemeny_kezelo()
{
    //lekerem a mezoket
    var sex = document.getElementById("sex") //id
    var rendezes_2 = document.getElementById("rendezes")
    var atlag = document.getElementById("atlag")
    var fialas_darab = document.getElementById("fialas_darab")

    if( tipus.value == "Koca")
    {
        document.getElementById("sex").disabled = true; //inaktív
        document.getElementById("rendezes").disabled = false; //aktív
        document.getElementById("atlag").disabled = false;
        document.getElementById("fialas_darab").disabled = false;

        var op = document.getElementById("rendezes").getElementsByTagName("option");
        // az option-et aktívra állítom
        op[3].disabled = false;
        op[4].disabled = false;
        op[5].disabled = false;
        op[6].disabled = false;
        op[7].disabled = false;
        op[8].disabled = false;
        op[9].disabled = false;
    }
}
```

4.5 ábra Szűrő funkcionál ha a típus Koca akkor több rendezési lehetőség közül lehet választani



A Szűrésnél a lekérdezésekben változókat használok az egyszerűsítés és a megvalósíthatóság érdekében. Ezek a változók a táblanevet, a fajtát, a típust jelölik a legtöbb esetben (lásd 4.6 és 4.7 ábra).

```
//harmadik rendezes: életkor szerint csökkenő
if( $rendezes=="Életkor szerint csökkenő" )
{
    $lekerdezes="SELECT $tibblev1.Sorszam AS Sorszam, $tibblev2.Koca_azonosito AS Azonosito, $tibblev1.Anya_azonosito AS
    Anya_azonosito, $tibblev1.Apa_azonosito AS Apa_azonosito, $tibblev1.Szuletesi_datum AS Szuletesi_datum, $tibblev1.Fajta AS
    Fajta, $tibblev1.Nem AS Nem, $tibblev1.Tipus AS Tipus, COUNT($tibblev2.Sorszam) AS Fialasok, AVG($tibblev2.Fialt_malacok) AS
    Atlagi, AVG($tibblev2.Felnevelt_malacok) AS Atlag2 FROM $tibblev1, $tibblev2 WHERE $tibblev2.Koca_azonosito=$tibblev1.
    Azonosito GROUP BY Koca_azonosito ORDER BY Szuletesi_datum DESC";
}
```

4.6 ábra Szűrő: a típus Koca a változók a táblanevet jelölik

```
$tabla=mysqli_query( $conn, " SELECT Sorszam, Azonosito, Anya_azonosito, Apa_azonosito, Szuletesi_datum, Fajta, Nem,
Tipus FROM $tibblev WHERE Nem LIKE '$sex' AND Fajta LIKE '$fajta' AND Tipus LIKE '$tipus' ");
```

4.7 ábra Szűrő-lekérdezés: A sex, fajta, típus változók a nemet, a fajtát és a típust jelölik

## 5. Fejlesztési lehetőségek

A weboldal alap funkciókra képes, így több fejlesztési lehetőséget lehetne megvalósítani:

- A bemeneti adatok helyességét ellenőrző rendszernek a fejlesztése
- A Szűrő funkció fejlesztése
- A felhasználóknak legyen külön egy bemutatkozó oldal, ahol egy rövid leírást adhatnak meg magukról, képeket tölthetnek fel a farmról, illetve ha rendelkeznek termékkel azt publikussá tudják tenni
- Az adatbázisban levő sertések családfájának/ágrajzának elkészítése
- A weboldal román nyelven is legyen elérhető
- Az adatok automatikus frissítése, például egy malacnak bejegyzett sertés x nap múlva legyen süldő, illetve y nap múlva legyen hízó
- A felhasználók számára legyen elérhető az adatbázisok letöltése egy gombnyomással PDF, Word, illetve Excel dokumentum formájában
- Értesítés küldés a felhasználónak a fialási időszakban

## 6. Szakirodalom

- Wikipedia  
[wikipedia.org](https://wikipedia.org)
- W3Schools  
[www.w3schools.com](https://www.w3schools.com)
- Stack Overflow  
[stackoverflow.com](https://stackoverflow.com)
- Horn Péter – Állattenyésztés