

WEBPROGRAMOZÁS III. GY.

(LBT_IM744G2)

Beadandó feladat dokumentáció

Kovács Norbert
ANOXWJ

2021. június 20.

Tartalomjegyzék

1. Weboldal koncepciója	3
1.1. Technikai részletek	3
1.2. Elképzelés	3
2. Adatbázis tervezése	3
2.1. ER modell	4
2.2. Relációs adatmodell	5
2.3. SQL parancsok	6
3. Összefoglaló	6

1. Weboldal koncepciója

Github link:

<https://github.com/Gellifree/bluepaw.git>

1.1. Technikai részletek

A beadandó feladatot, az órán is alkalmazott *xampp* szerverrel kívánom megvalósítani, *apache netbeans* fejlesztőkörnyezettel. A Mysql kezelését a videóban is használt, *phpmyadmin* oldalon intézem. Nem fontos részlet, a megvalósítás szempontjából, de a Windows operációs rendszer, virtuális gépként fut.

Követelmény: „A beadandó feladat PHP MVC keretrendszerben készül!”

- OS: Windows 10 Pro (21H1)
- Kernel: 10.0.19041.1023
- Mysql: 10.4.19-MariaDB
- PHP: 8.0.6
- Apache Netbeans IDE: 12.3
- CodeIgniter: 3.1.11

1.2. Elképzelés

A weboldal egy *fiktív* szervezet köré épül, aminek a feladata az *állategészségügy*, és *állatvédelem*. A fiktív szervezetnek több városban lesznek épületei országszerte. Egy városban több különféle épület is tartozhat a szervezethez. Például tartozhat állatmenhely, ahol kutyákat tárolnak, állatorvosi rendelő vagy éppen raktárépület.

Hasonlóképpen az egyetemi példafeladathoz, ahol

Kampusz -> Épület -> Terem

Pl.: Eger -> C épület -> 124-es terem,

felosztás található, itt is három rétegre kívánom felosztani a problémát, mégpedig:

Régió -> Épület -> Iroda

Pl.: Miskolc -> Orvosi rendelő -> 1-es vizsgáló

A fiktív szervezetet, az előző beadandóm állatotthonából fejleszttem tovább, részben idő spórolás céljából, részben a hasonló koncepcióból adódóan.

Szervezet neve: **Kék mancs**.

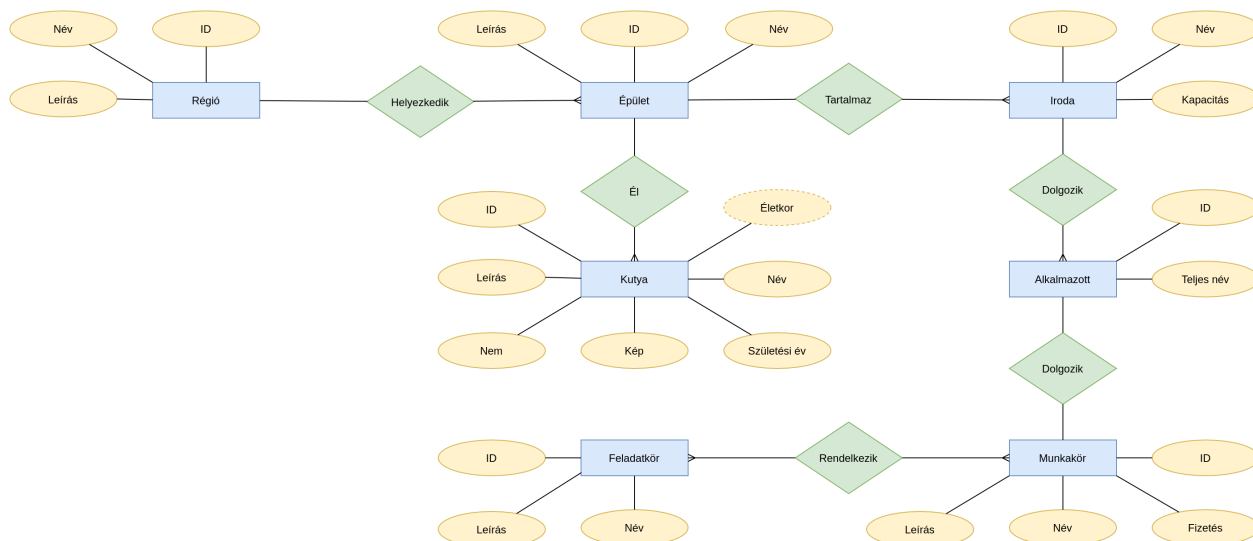
2. Adatbázis tervezése

Az adatbázis során a következőkben tervezett entitásokon kívül, még számos entitással lehetne bővíteni az adatbázist, viszont így is 7 tábla születik, a kapcsoló tábla figyelembe vétele nélkül.

2.1. ER modell

A pluszba kapott egyedi azonosító mezőket, és az idegen kulcsokat nem tüntetjük fel a tervezés során.

1. Régió
Egy Régió eltárolásához, elegendő a régió *neve* (ami gyakorlatilag a város neve), és egy *leírás* mező.
2. Épület
Az épület entitás kapcsán, elég tárolnunk az épület *nevét*, és *leírását*.
3. Iroda
Egy épület alegysége, az iroda entitás lesz. Elegendő tárolnunk a *nevét*, és a *kapacitását*.
4. Kutya
Egy kutya esetén, tárolnunk kell a *nevét*, *leírását*, *születési évét*, *nemét* és egy hozzá tartozó *képet*. A kutya *életkora*, származtatott tulajdonság lesz.
5. Alkalmazott
Az alkalmazottak esetén, tároljuk el az alkalmazott teljes *nevét*.
6. Feladatkör
A feladatkörnek *tároljuk* a nevét és *leírását*.
7. Munkakör
A munkakör kapcsán, tároljuk a *nevét*, a *leírását*, és a hozzá tartozó *fizetést*.

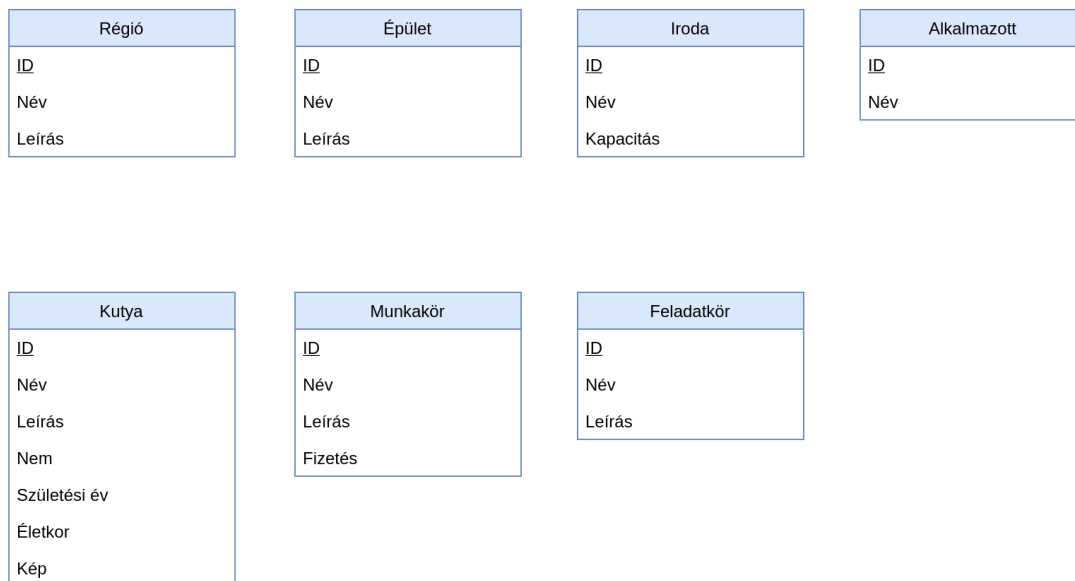


Az ER modell

Követelmény: „A megvalósítandó projekthez tartozó adatmodell legalább 5 forgalmi adattáblát tartalmaz (a külső modulok által használt táblákon kívül). Az adatmodell legalább 3. normálformában van és modellben megtalálható az egy-több, valamint a több-több típusú kapcsolat is. A technikai okokból létrehozott táblák (kapcsolótáblák) nem tekintendők forgalmi táblának.”

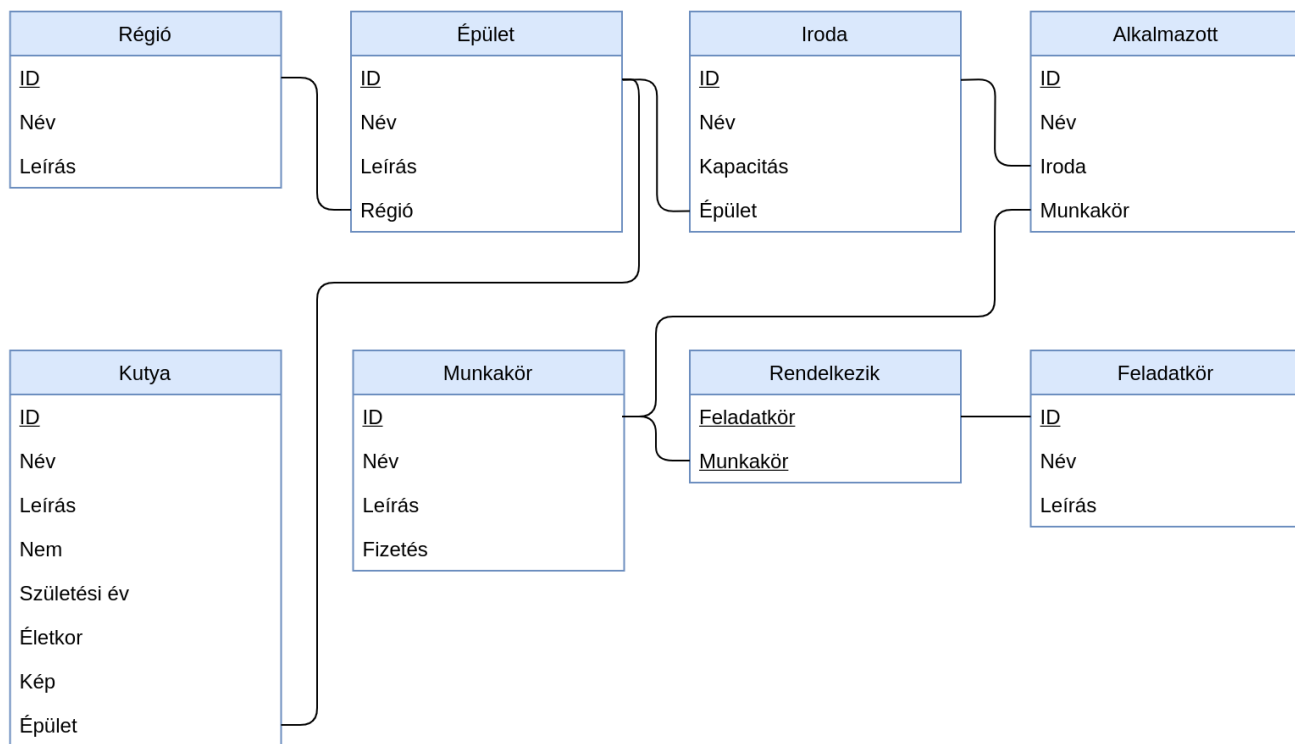
2.2. Relációs adatmodell

Az elkészült modell alapján, a következő táblák jöttek létre.



Az ER modell

Hozzuk létre a szükséges kapcsolótáblákat, és illesszük be az idegen kulcsokat, majd jelöljük a kapcsolatokat.



Az ER modell a kapcsolatokkal

2.3. SQL parancsok

Az *SQL* parancsok megtalálhatóak az *application/create.sql* fájlban. Ebben, minden *SQL* parancs megtalálható, amit manuálisan futtattunk a *phpmyadmin* oldalon keresztül. A végleges adatok természetesen már nem *SQL* parancsokkal kerültek fel a weboldalra, hanem a grafikus felületen keresztül, így azok itt nem rögzítettek.

3. Összefoglaló

Összességében, a projekt sok hiányossággal rendelkezik. A követelményekből, a fájl feltöltés, megvalósításra került, és integrálva lett, a fájlok letöltésének funkciója, ugyan technikailag megtalálható a projekt forrásállományai között, az semmilyen formában nem lett integrálva a weboldallal, és nem használható, így ez nem tekinthető megvalósítottnak.

A nézetek reszponzívak ugyan, de a táblázatok eseténél a reszponzivitás nagyon kis ablakméretnél megtörhet.

A csoportkezelés technikailag megvalósul, és minden Controller eseténél megtalálható a csoport ellenőrzése, a megfelelő műveleteknél. Általánosan kifejezve a *Controller* nevét követi a *manager* kifejezés.

A gyakorlatban még számos egyéb kényelmi probléma vetül fel. Egy adott munkakörhöz hozzá lehet rendelni bármennyi feladatkört, és ellenőrizve is vannak, egyedi validációs függvénnyel a duplikációk, de a felületen, nincs átadva a hozzáadó nézetnek, a munkakör azonosítója, így azt is ki kell választani, amikor feladatkört rendelünk hozzá. Ez kifejezetten kényelmetlen, és megoldásra szorulna, a munkakör paraméterének átadásával.

Az adatbázis felépítése során úgy építettük fel az adatokat, hogy egy alkalmazott egy irodához, nem épülethez rendelt. Ez alapesetben rendben is volna, egészen addig, amíg nem olyan munkakörről van szó, ami nem értelmezhető csak irodára. Például egy takarító alkalmazottat, nem irodához kellene értelmeznünk, hanem épülethez. Bizonyos munkakörök esetében, viszont az iroda megközelítés áll helyt, pl egy adminisztrátori munkakörben. Ennek feloldásához valószínűleg újra kellene tervezni az adatbázist.

Ezenkívül praktikus volna, különféle lekérdezéseket megvalósítani, a listázó nézetek praktikusabbá tétele érdekében. A szűrés funkciója, miszerint egy adott tulajdonság alapján szűrjünk, kifejezetten hasznos lenne. Pl csak azokat az épületeket listázza ki, amik egy adott régióhoz tartoznak.

Ezenkívül, egy egyedi rendezési szempont elkészítése is hasznos lenne. Miszerint, a listázó nézet, valamilyen oszlop alapján rendezett, és ezt a felhasználó változtathatja. Ezt véleményem szerint, úgy lehetne megoldani, hogy a listázó nézetet, elválasztom az egyedi nézettől, és a listázó nézet paraméterét, a rendezés szempontjaként értelmezném. Ezáltal, a megfelelő helyre, a megfelelő id-vel rendelkező linkek alapján, át tudnánk adni a modellen keresztül egy olyan lekérdezést, ahol a rendezés szempontja, a paraméterben kapott mező. Természetesen ebben az esetben, az egyedi nézethez rendelnünk kell egy új függvényt.