

Városi tömegközlekedés Projekt

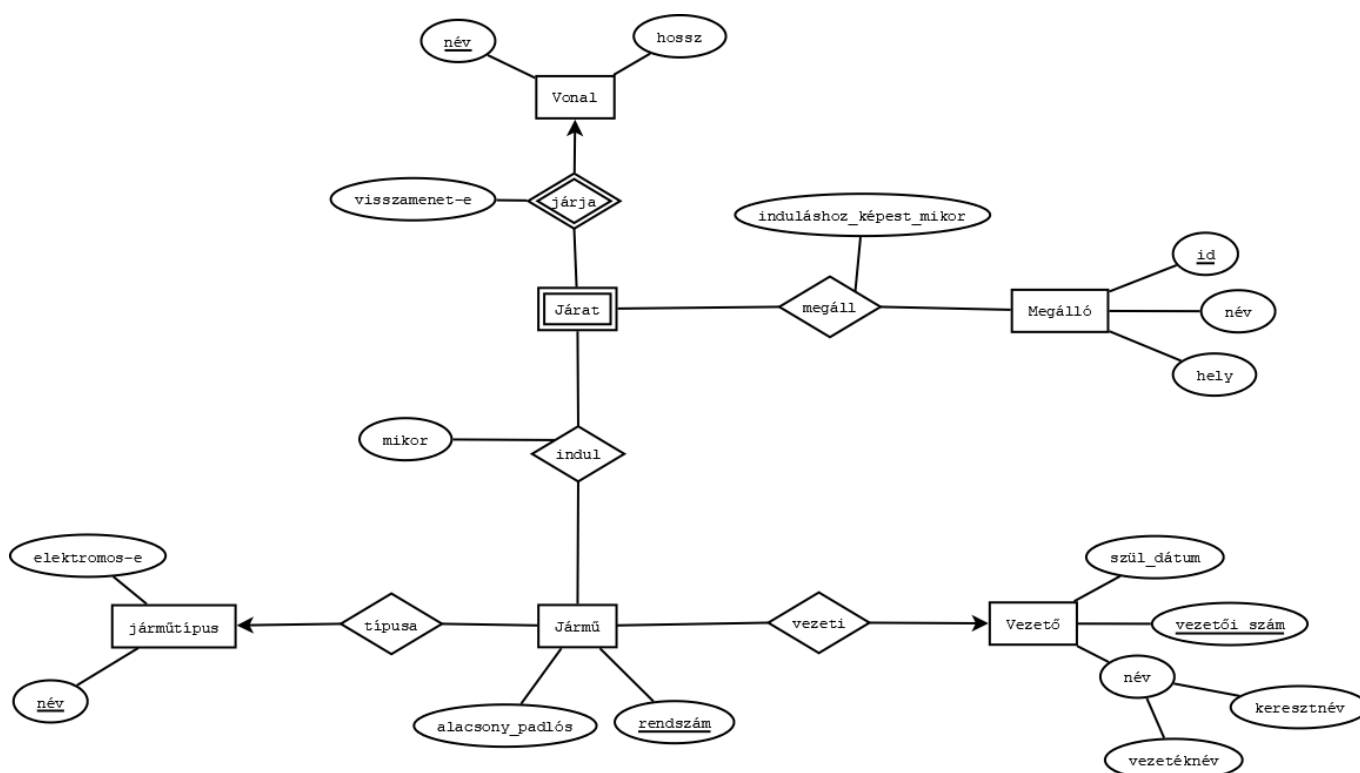
Adatbázisok 2020. Löffler Levente

1. Ismertető

Az adatbázis egy város tömegközlekedését reprezentálja. A felhasználók indulási és érkezési állomások szerint információt szerezhetnek különböző járatokról. Ezen információk közé tartoznak pl.: indulás ideje, érkezés ideje, jármű típusa, mozgáskorlátozottsága felkészültség, vonal hossza.

Az adatbázist egy alkalmazáson keresztül lehet elérni. Ez rendelkezik megfelelő grafikus felülettel a fentebb említett adatok böngészésére. Van továbbá lehetőség regisztrációra, a regisztrált felhasználók további információkat is megtekinthetnek pl.: vezető neve, jármű rendszáma. A megfelelő beosztású dolgozók adminisztrátori jogosultsággal rendelkező fiókot kapnak, ami az adatbázis módosítására is lehetőséget ad.

2. Az adatbázis egyed-kapcsolat diagramja



3. Az diagram sémákká való leképezésének folyamata

Jelölések: a sémák kulcsait **félkövéren**, a külső kulcsokat pedig **dőlten** jelölöm.

A diagramon látható egyedek a következők: Vezető, Jármű, Járműtípus, Járás, Vonal, Megálló.

- Egy jármű jelképez egy konkrét, rendszámmal ellátott fizikai járművet.
- A járműtípusok diktálják a járművek típusának értéktartományát.
- A vonal jelképezi az útvonalat, amelyen a járművek járnak.
- A járat reprezentálja a járművek összességét, amelyek az adott vonalon járnak.
- A megálló egy fizikai megállóhely.

A vezető, a jármű, a vonal, és a megálló egyértelműen leképezhetőek sémákká:

- Vezető(**vezetői_szá**m, vezetéknév, keresztnév, szül_dátum)
- Jármű(**rendszám**, alacsony_padlós)
- Járműtípus(**név**, elektromos)
- Vonal(**név**, hossz)
- Megálló(**id**, név, hely)

A járat egy gyenge egyed. Egy járat csak egy vonalon megy, illetve egy vonalon 2 féle járat jár: egy a vonal A végállomásából a B-be, és egy visszafelé. Így egy járatot azonosíthatunk az alapján, hogy melyik vonalon, milyen irányba megy. A vonal neve külső kulcs lesz:

- Járat(**von**al_név, **vi**sszamenet)

Jöjjenek a kapcsolatok. Ezek a következők: a vezetők vezetik a járműveket, a járművek felvesznek egy típust maguknak és elindulnak járatként bizonyos időpontokban, a járatok pedig megállnak megállóknál bizonyos időpontokban.

- Vezeti(**vezetői_szá**m, **rendszám**)
- Indul(**rendszám**, vonal_név, visszamenet, **mikor**)
- Megáll(**von**al_név, **vi**sszamenet, **megálló_id**, induláshoz_képest_mikor)
- Típusa(**rendszám**, **típus_név**)

Ezek után végezhetünk összevonást a Vezeti és Típusa 1:N kapcsolatokon:

- Jármű(**rendszám**, alacsony_padlós, *típus_név*, vezetői_szá

Ezzel a teljes egyed-kapcsolat diagram le van képezve sémákká. A végleges sémák:

- Vezető(**vezetői_szá**m, vezetéknév, keresztnév, szül_dátum)
- Járműtípus(**név**, elektromos)
- Jármű(**rendszám**, alacsony_padlós, *típus_név*, vezetői_szá
- Vonal(**név**, hossz)
- Megálló(**id**, név, hely)
- Járat(**von**al_név, **vi**sszamenet)
- Indul(**rendszám**, vonal_név, visszamenet, **mikor**)
- Megáll(**von**al_név, **vi**sszamenet, **megálló_id**, induláshoz_képest_mikor)

Ezekon kívül az adatbázis rendelkezik az alábbi egyszerű user sémával:

- User(**username**, password, email, admin) Ez az alkalmazásban regisztrált felhasználókat reprezentálja. Az EK diagramon nem szerepel, és a későbbiekben nem is részletezem, mert triviális, és nincs szoros kapcsolatban az adatbázis témájával, csupán az alkalmazás működéséhez szükséges.

4. A sémák normalizálása

1NF

A sémák már 1. normálformában vannak, hiszen nincs többértékű attribútum, az összetett attribútumokat pedig eleve részekre bontva képeztem le a sémákban.

2NF

Az egyedek sémái (a járatot leszámítva) mind egyelemű kulcsokkal rendelkeznek, így azok 2. normálformában vannak. A járatnak nincsenek más attribútumai a kulcsain kívül, így az is.

Az Indul vonalnév és visszamenet attribútumai teljesen függnék a kulcstól, hiszen ugyanaz a jármű több vonalon is indulhat különböző időpontokban, ráadásul más irányokba is, illetve több jármű is indulhat egy időpontban, de más vonalakon és/vagy irányba. Végezetül a Megáll induláshoz_képest_mikor attribútuma is teljesen függ a séma kulcsától, hiszen más vonanalon közlekedő járművek más megállóknál állnak meg, a vonal iránya pedig befolyásolja, hogy az adott megállóhoz induláshoz képest hányadik percben ér. Így tehát az összes séma megfelel a 2. normálformának.

3NF

Több másodlagos attribútummal csak a Vezető, Jármű, Megálló, és Indul sémák rendelkeznek. A vezető esetében a neve és a születési dátuma között semmilyen összefüggés nincs. Ugyanez igaz a jármű típusára, padlószintjére, és vezetői számára. Egy megálló neve és helyszíne között sincsen kapcsolat. Az indulás vonala nem határozza meg az irányát, és az irány sem határozza meg a vonalat. Ezek így 3. normálformában vannak, ahogy a Járműtípus, Járát, Vonal, és Megáll sémák is, hiszen azokban nincs egynél több másodlagos attribútum, így nincs amik közt függés alakulhatna ki.

5. Az alkalmazás ismertetése

Az alkalmazás Python nyelven íródott, és a mysql.connector, tkinter, illetve tkinter.ttk csomagokat használja. Bár a program nincsen folyamatos kapcsolatban az adatbázissal, a futtatáshoz elengedhetetlenül szükséges az adatbázis elérés, különben egyes háttér folyamatok (pl.: megjelenő választási lehetőségek listájának frissítése) sikertelenül fejeződhetnek be.

6. Az alkalmazás használata

Az alkalmazásnak 3 használati esete van. Ebben a részlegben ezeket részletezem.

Vendég felhasználó

Vendég felhasználónak minősül az, aki nincs bejelentkezve. Az ilyen felhasználók menetrend keresésre használhatják az alkalmazást. Az alkalmazás indítása után a fejlécben található megálló kereső segítségével kereshetnek járatokat, amelyeken utazva eljuthatnak az induló megállóikból a célállomásukig. Az egyes járatokról részletesebb információt is kaphatnak, ha megnyomják a "Részletek" feliratú gombot. Ez az adott járat kiválasztott időpontbeli menetét listázza. Innen leolvashatják a felhasználók a járat végállomásait, és hogy mikor ér az egyes állomásokhoz. Az egyes állomások mellett szereplő "Részletek" gomb megnyomásával pedig további információkat kaphatnak a kiválasztott megállóról (milyen járatok és mikor állnak meg ott). A különböző oldalak között a bal felső sarokban található "Előre", illetve "Vissza" gombokkal navigálhatnak a felhasználók. A "Főoldal" gomb mindig a kezdőoldalra visz. A vendég felhasználók ingyenesen regisztrálhatják magukat a főoldalon található "Regisztráció" gomb megnyomásával megjelenő űrlap kitöltésével és elküldésével. Regisztráció után a bejelentkezésre is itt van lehetőség, a "Bejelentkezés" gomb megnyomásával.

Regisztrált felhasználó

A regisztrált felhasználók további információk szerezhetnek az egyes járatokról. A járatokat részletező oldalon 3 új mező fogadja őket: a vezető neve, a vonal hossza, illetve hogy az adott jármű alacsony padlós-e.

Adminisztrátor

Egy admin képes mindenre, amire az előző két csoport képes, ezen felül pedig lehetősége van az adatok részletes, listaszerű szerkesztésére, új adatok felvitelére, illetve meglévő adatok törlésére. A főoldalon elérhető számukra három új gomb, az előbb említett három új funkcióhoz egy-egy. Az opció kiválasztása után mindhárom esetben a kategóriaválasztás a következő lépés. Ezek az alábbiak:

- Vonal
- Megálló
- Vezető
- Járműtípus
- Jármű
- Járat

Új adat felvitele esetén a választás után egy űrlap jelenik meg, ahol az adminisztrátor megadhatja az új adatokat. Adatmódosítás esetén először ki kell választania a módosítandó adatot egy listából, majd ezt követően jelenik meg egy előre kitöltött űrlap (hasonló az új adat űrlaphoz), amelyen keresztül végezhető el a frissítés. Törlés esetén csak az azonosítót kell kiválasztani egy listából. Figyelem! Egy adat törlése maga után vonja a tőle függő adatok törlését is!

7. Felhasznált eszközök

A szoftver Visual Studio Code-ban íródott, a Microsoft Python illetve Pylance kiegészítőit használva. Az alkalmazás Python nyelven fut, továbbá a tkinter, tkinter.ttk, és mysql.connector csomagokat használja. Az EK diagram a Dia nevű programmal készült. Az adatbázis az XAMPP szoftvercsomag MariaDB modulját használva készült. Az adatfeltöltés eleinte a PHPMyAdmin felületen keresztül történt, majd később a már kész és futó Python alkalmazással fejeződött be.

8. További megjegyzések

Az adatbázisban található 2 felhasználó a különböző jogosultságok teszteléséhez. Adminisztrátori felhasználó:

- Név: admin
- Jelszó: admin

Általános felhasználó:

- Név: user
- Jelszó: user