Adatbázis

Egy játékszoftver működésének kiemelkedően fontos eleme egy megbízható, jól megtervezett adatbázis, tervezés hiányában ugyanis a szoftver karbantartása körülményesebbé válhat, bővíthetősége megnehezül.

A mi adatbázisunkat a Laravel keretrendszer beépített adatbáziskezelő megoldásaival hoztuk létre, ugyanis a migrációk és az adatbázis-seederek rendkívül egyszerűvé teszik a fejlesztést bárhonnan.

1. Hogyan jön létre az adatbázis?

Egy lokális adatbázist egy újonnan beüzemelt fejlesztői számítógépen nagyon egyszerűen létre lehet hozni az előbb említett eszközökkel. Miután elindítottuk az SQL szerverünket (általában xampp) és megnyitottuk a Laraveles projektet, azután szimplán nyitnunk kell egy terminálablakot, majd beírni a ‘php artisan migrate:refresh –seed’ parancsot.

Ez által létrejön az SQL szerveren egy ‘houseofswords’ nevű adatbázis. Az adatbázis létrejöttekor a táblák az adatbázis-seedereknek köszönhetően automatikusan feltöltődnek tesztadatokkal, így például az ‘admin’ nevű felhasználó már egyből -- egy játszható várossal, épületekkel, és kikutatott egységekkel – létrejön.

1. Mi a helyzet a központi szerveren lévő adatbázissal?

A képen szöveg látható

Automatikusan generált leírásA központi szerveren lévő adatbázis is hasonlóképpen módosítható, annyi különbséggel, hogy az adatbázishoz tartozó környezeti változókat a ‘.env’ fájlban át kell írnunk a szerver adataira. A képen látható a különbség a változók között mindkét esetben. Ha lokális adatbázist szeretnénk használni, akkor a fájlban a szerver adataihoz tartozó 6 sort ki kell kommentezni, és vice versa.

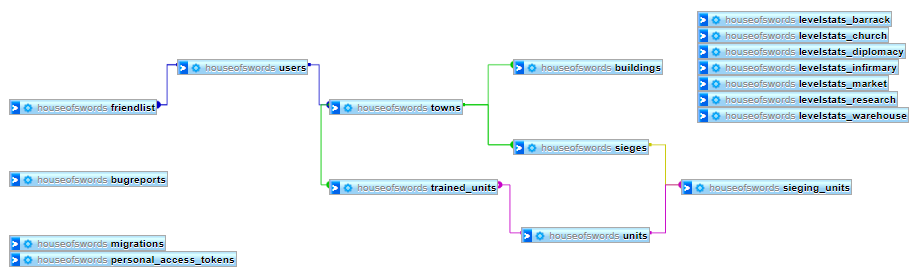
1. Hogyan lehet megtekinteni, illetve módosítani az adatbázist és a benne lévő adatokat?

Lokális adatbázis esetében egyszerűen – DBForge vagy phpMyadmin segítségével.

A szerveren lévő adatbázist a <https://db.houseofswords.hu> domain című oldalon lehet elérni, a bejelentkezési adatok megadása után. A phpMyadmin felületen a szokásos módon lehet felvenni, módosítani és törölni táblákat és rekordokat.

1. Milyen táblák, modellek és kapcsolatok vannak az adatbázisban?

Az alábbi képen látható az adatbázisunk összes táblája, illetve a köztük lévő kapcsolatok. Az ábra a phpMyadmin felület vizuális szerkesztőjéből lett fotózva.

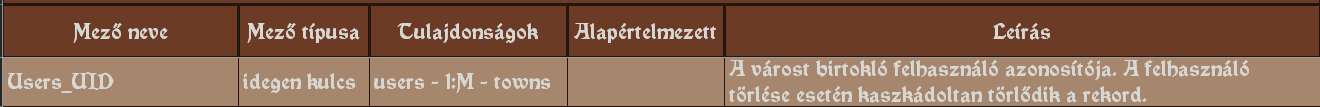


A dokumentációhoz tartozik egy .xlsx kiterjesztésű állomány ‘Modellek gyűjteménye’ néven. Ebben a táblázatban minden lapon egy-egy adatbázis-modell mezői, illetve kapcsolatai láthatóak, az alábbi módon. (Megtekintéshez a mellékelt betűtípus, a Vinque Rg telepítése ajánlott, de nem kötelező.)

A képen asztal látható

Automatikusan generált leírás

Amennyiben a tábla egyik mezője kapcsolódik egy másikhoz, úgy a tulajdonságok mezőben az is meg lesz jelölve. Például az egy-a-többhöz kapcsolat miatt a ‘towns’ táblában lévő ‘Users\_UID’ mező, ami annak a felhasználónak az azonosítójára mutat, akié a város.



A vizuális szerkesztőből fotózott képen látható kapcsolatok nem minden tábla között állnak fent. Néhol olyan egyedi, logikai kapcsolat áll fent két tábla között, amit a szabványos SQL szabályaival nem lehet megvalósítani. Például a ‘bugreports’ tábla arra szolgál, hogy a felhasználók által jelentett hibaüzeneteket tárolja. Ha a felhasználó a rögzítés időpontjában be volt jelentkezve, akkor elmentjük az email címét, ami alapján beazonosítható, hogy ki jelentette a hibát. Ellenkező esetben ha nincs bejelentkezve a felhasználó, akkor ‘null’ érték kerül az email cím mezőbe.

Az SQL nem engedi olyan kapcsolatok létrejöttét, ahol lehet ‘null’ az idegen kulcs, ezért a kapcsolat csak ‘logikai’ lesz, és a Laravel modell osztályában valósítható meg.

A ‘Laravel kapcsolatok’ részben a táblázat alsó részén az kerül leírásra, hogy egy modell objektumnak milyen függvénye van, ami megvalósít egy-egy ilyen kapcsolatot.



Például egy ‘user’ típusú objektumnak Laravelen belül van egy ‘bugreports’ nevű függvénye, ami visszaadja az összes olyan rekordot a ‘bugreports’ táblából, ahánynál az email cím mező egyezik a felhasználó email címével.

Ilyen logikai kapcsolat áll fent még például a ‘buildings’ és a ‘levelstats’ táblák között. Egy ‘building’ objektum csak akkor kapcsolódik egy ‘levelstats\_épületTípus’ táblához, ha a ‘building’ objektum típus mezője egyezik az ‘épületTípus’-sal. Vagyis egy templom típusú épület a ‘levelstats\_church’ táblához kapcsolódik, míg egy raktár típusú a ‘levelstats\_warehouse’ táblához.

A táblázatban több információ van az egyes modellekről, mezőikről és kapcsolataikról.