Az adatbázis ER diagrammja

A képen szöveg, diagram, Tervrajz, Műszaki rajz látható

Automatikusan generált leírás

DatabaseContext

* Az EntityFramework adatbázist reprezentáló objektum. Az IdentityDbContext leszármazottja, mivel a program IdentiyUsereket tartalmaz. Minden Entity-hez tárol egy DbSet típusú propertyt, melyeket a OnModelCreating metódusban inicializál. Ezen kívül van még egy dictionary propertyt EntityTables néven, mely tartalmazza az összes DbSet-et. A GenericCrudOperator ennek a segítségével tudja kikeresni az adott típushoz tartozó DbSet-et.

DbInit

* Mindössze egy statikus metódust tartalmaz Init néven, mellyel az adatbázist lehet inicializálni dummy adatokkal. Minden inicializálás előtt kitörli a tárolt adatokat.

User

* Egy Usert reprezentáló entity. Nem tartalmaz egyedu property-t, csak azokat, melyeket az IdentityUser nevű osztálytol örököl. Ez az ASP.NET Identity-be lévő osztály felel a felhasználó tábla oszlopaiért, a hitelesítésért, jelszó hashelésért.
* Dto-k:
  + UserRegisterDto 🡪 Egy felhasználó regisztrálásához használható Dto.
  + UserLoginDto 🡪 Egy felhasználó belépéséhet használható Dto.
  + UserDatasDto 🡪 Egy felhasználó adatai lekérdezéséhez használható Dto.
  + UpdateUserDto 🡪 Egy felhasználó adatainak frissítéséhez használható Dto.

Owner

* Egy tulajdonos felhasználó és egy éttermet összekötő kapcsolótáblát reprezentáló entitás.
* Properties:
  + Id (string) 🡪 Kulcs.
  + UserId (string) 🡪 A felhasználó id-je, idegen kulcs.
  + User (User) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + RestaurantId (string) 🡪 Az étterem id-je, idegen kulcs.
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property.

Admin

* Egy Admint reprezentáló tábla entitása:
* Properties:
  + Id (string) 🡪 Kulcs.
  + UserId (string) 🡪 A felhasználó id-je, idegen kulcs.
  + User (User) 🡪 Objektum szinten összeköttetést biztosító navigation property.

Restaurant

* Egy éttremet reprezentáló tábla entitása:
* Properties:
  + Id (string) 🡪 Kulcs.
  + Menu (Menu) 🡪 Menu entitás objektum navigation property
  + MenuId (string) 🡪 A menu property Id-je
  + Name (string) 🡪 Az étterem neve
  + Description (string) 🡪 Az étterem leírása
  + Label (string) 🡪 Az étterem „szolgenje”
  + City (string) 🡪 Város, ahol az étterem található
  + Street (string) 🡪Az utca, ahol az étterem található
  + HouseNumber (int) 🡪 A házszám
  + PostCode (int) 🡪 Az irányítószám
  + PhoneNumber (string) 🡪 Az étterem telefonszáma
  + NumOfFreeSeats (int) 🡪 Az étteremben található szabad ülőhelyek száma.
* Dto-k
  + RestaurantDataDto 🡪 Egy étterem adatainak lekérdezéséhez használható Dto.
  + CreateRestaurantDto 🡪 Egy étterem létrehozására alkalmas Dto.
  + UpdateRestaurantDto 🡪 Egy étterem adatainak frissítéséhez használható Dto.

Reservation

* Egy foglalást reprezentáló entitás.
* Properties:
  + Id (string) 🡪 Kulcs.
  + ReserverId (string) 🡪 A felhasználó id-je, idegen kulcs.
  + Reserver (User) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + RestaurantId (string) 🡪 Az étterem id-je, idegen kulcs.
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + DateTime (string) 🡪 A foglalás időpontja
  + NumOfPeople (int) 🡪 A lefoglalt helyek/személyek száma
  + Lenght (int) 🡪 A foglalás hossza
  + Comment (string) 🡪 Megjegyzés a foglaláshoz
* Dto-k:
  + ReservationDto 🡪 Egy foglalás adatainak lekéredezeséhez használható Dto.
  + CreateReservationDto 🡪 Foglalás létrehozásához használható Dto.

Review

* Egy étterem értékelést reprezentáló entitás.
* Properties:
  + Id (string) 🡪 Kulcs.
  + UserId (string) 🡪 A felhasználó id-je, idegen kulcs.
  + User (User) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + RestaurantId (string) 🡪 Az étterem id-je, idegen kulcs.
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + Rating (int) 🡪 Az étterem értékelése számban kifejezve
  + Description (string) 🡪 Az értékeléshez tartozó szöveges leírás
* Dto-k
  + ReviewDto 🡪 Egy értékelés adatainak lekérdezéséhez használható Dto.
  + CreateReviewDto 🡪 Egy étterem létrehozására alkalmas Dto.

LikedRestaurant

* Egy étterem kedvelést reprezentáló entitás.
* Properties:
  + Id (string) 🡪 Kulcs.
  + UserId (string) 🡪 A felhasználó id-je, idegen kulcs.
  + User (User) 🡪 Objektum szinten összeköttetést biztosító navigation property.
  + RestaurantId (string) 🡪 Az étterem id-je, idegen kulcs.
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property.
* Dto-k:
  + LikedRestaurantDto 🡪 Egy étteremkedvelés lekérdezéséhez/létrehozásához használható Dto.

Menu

* A Menu osztály egy étterem menüjét reprezentálja
* Properties:
  + Id (string) 🡪 Elsődleges kulcs
* Konstruktor:
  + Menu() 🡪 Alapértelmezett konstruktor

Category

* A Category osztály egy kategóriát reprezentál az éttermi menün belül
* Properties:
  + Id (string) 🡪 Elsődleges kulcs
  + MenuId (string) 🡪 Idegen kulcs, egyedi azonosító annak a menünek, amelyhez ez a kategória tartozik
  + Menu (Menu) 🡪 Objektum szinten összeköttetést biztosító navigation property
  + Name (string) 🡪 A kategória neve
* Konstruktor:
  + Category() 🡪 Alapértelmezett konstruktor
  + Category(Menu menu, string name) 🡪 Paraméterezett konstruktor, amely inicializál egy új kategóriát a megadott menüvel és névvel
* Dto-k:
  + CreateCategoryDto 🡪 új kategória létrehozásához használható Dto
  + CategoryDto 🡪 kategória információk lekérdezéséhez használható Dto

Food

* A Food osztály egy étel entitást reprezentál, amely egy adott kategóriához tartozik egy éttermi menüben
* Properties:
  + Id (string) 🡪 Elsődleges kulcs
  + CategoryId (string) 🡪 Idegen kulcs, egyedi azonosító annak a kategóriának, amelyhez ez az étel tartozik
  + Category (Category) 🡪 Objektum szinten összeköttetést biztosító navigation property
  + Name (string) 🡪 Az étel neve
  + Description (string) 🡪 Az étel leírása
  + Price (int) 🡪 Az étel ára
  + Image (byte[]) 🡪 Az ételhez tartozó kép, lehet null
* Konstruktor:
  + Food() 🡪 Alapértelmezett konstruktor
  + Food(Category category, string name, string description, int price) 🡪 Paraméterezett konstruktor, amely inicializál egy új ételt a megadott kategóriával, névvel, leírással és árral
  + Food(Category category, string name, string description, int price, byte[] image) 🡪 Paraméterezett konstruktor, amely inicializál egy új ételt a megadott kategóriával, névvel, leírással, árral és képpel
* Dto-k:
  + FoodDto 🡪 az étel információk lekérdezéséhez használható Dto
  + CreateFoodDto 🡪 új étel létrehozásához használható Dto

Table

* A Table osztály egy éttermi asztalt reprezentál
* Properties:
  + Id (string) 🡪 Elsődleges kulcs
  + RestaurantId (string) 🡪 Idegen kulcs, az étterem egyedi azonosítója, amelyhez az asztal tartozik
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property
  + NumOfSeats (int) 🡪 Az asztalnál található ülőhelyek száma
  + IsReserved (bool) 🡪 Jelzi, hogy az asztal foglalt-e
* Konstruktor:
  + Table() 🡪 Alapértelmezett konstruktor
  + Table(Restaurant restaurant, int numOfSeats) 🡪 Paraméterezett konstruktor, amely inicializál egy új asztalt a megadott étteremmel és ülőhelyek számával
* Dto-k:
  + CreateTableDto 🡪 új asztal létrehozásához használható Dto
  + TableDto 🡪 asztal információk lekérdezéséhez használható Dto

RestaurantOpeningHours

* A RestaurantOpeningHours osztály egy étterem nyitvatartásit reprezentálja
* Properties:
  + Id (string) 🡪 Elsődleges kulcs
  + RestaurantId (string) 🡪 Idegen kulcs, az étterem egyedi azonosítója, amelyhez a nyitvatartási idő tartozik
  + Restaurant (Restaurant) 🡪 Objektum szinten összeköttetést biztosító navigation property
  + DayName (string) 🡪 A nap neve, amelyre a nyitvatartási idő vonatkozik
  + OpeningHour (string) 🡪 Az étterem nyitási ideje az adott napon
* Konstruktor:
  + RestaurantOpeningHours () 🡪 Alapértelmezett konstruktor
  + RestaurantOpeningHours(Restaurant restaurant, string dayName, string openingHour) 🡪 Paraméterezett konstruktor, amely inicializál egy új nyitvatartási időt a megadott étteremmel, nappal és nyitási idővel
* Dto-k:
  + OpeningHourDto 🡪 a nyitvatartási idő információk lekérdezéséhez használható Dto