

Baza de date .NET si Api

MyPhotos

Filoş Gabriel



Faculty of Computer Science
University Alexandru Ioan Cuza Iași
Romania
16.03.2020

Cuprins

1	Schema baza de date	2
2	Api-ul bazei de date	4
3	Declaratie	6

1 Schema baza de date

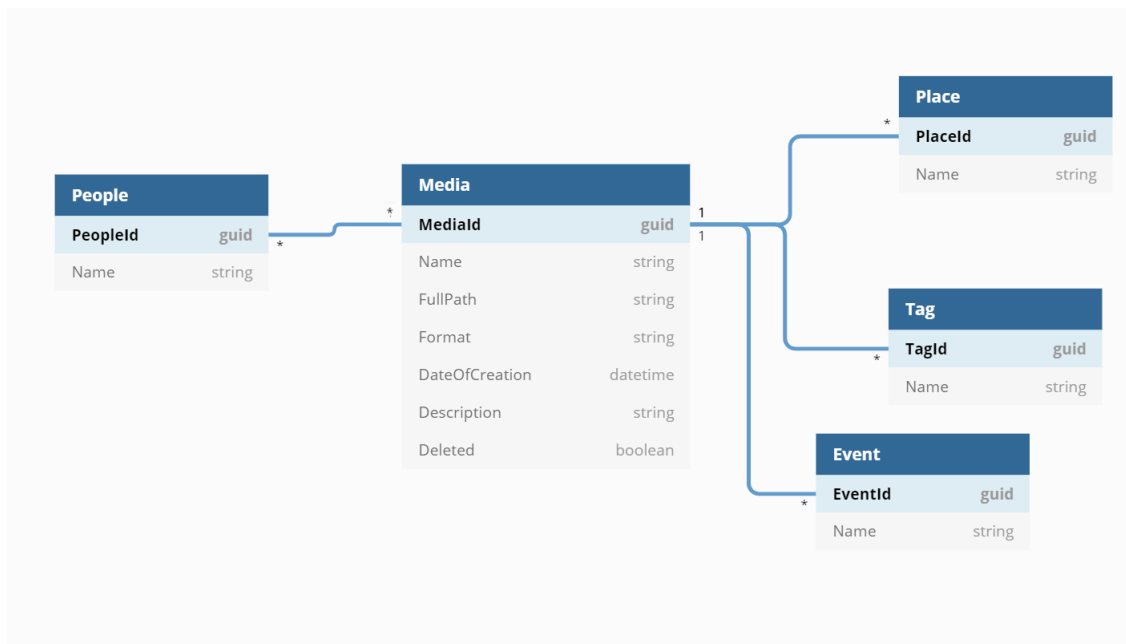


Figura 1: Baza de date

Descrierea tabelor este urmatoarea:

1. Media : aceasta tabela contine informatile de baza ale fisierelor media
2. Events: aceasta tabela contine numele evenimentelor pentru fiecare fisier media
3. Tag: aceasta tabela contine tagurile reprezentative pentru fiecare fisier media salvat
4. People: in aceasta tabela se vor memora numele fiecărei persoane care apare intru-un fisier media
5. Place: aceasta tabela contine informatii despre locurile in care s-au creat fisierele media

Tabela 1: Tabela People

PeopleId	Guid	indentificatorul unic al persoanei
Name	string	Numele persoanei

Tabela 2: Tabela People

MediaId	Guid	identificator unic fisier media
Name	string	Numele dat de catre utilizator fisierului media salvat in baza de date
FullPath	String	Calea completa a fisierului media
Forma	string	Formatul fisierului precum .mp4, .jpeg, etc
DateOfCreation	datetime	Data crearii fisierului, este luata din metadata fisierului
Description	string	Descrierea data de catre utilizator
Deleted	bool	Camp care marcheaza daca a fost sters

Tabela 3: Tabela place

PlaceId	Guid	identificator unic loc
Name	string	numele locului dat de utilizator

Tabela 4: Tabela Tag

PlaceId	Guid	identificator unic tag
Name	string	numele proprietati specifice date de catre utilizator

Tabela 5: Tabela Event

PlaceId	Guid	identificator unic eveniment
Name	string	numele evenimentului dat de catre utilizator

Am ales sa structurez asa tabelele deoarece am considerat ca ar fii mai bine ca unele elemente(precum Persoane, Tag-uri, Evenimente si Locuri) sa apara o singura data in baza de date. Un exemplu al motivatiei alegeri acestei decizi este urmatorul: o persoana poate aparea in mai multe fisiere media si ar fii mult mai eficient sa o memoram o singura data in baza de date decat sa o memoram de fiecare data intr-o coloana al tablei media. Aceasi motivatie se aplica si pentru tabelele Place, Tag, Event.

As fii putut merge si cu o abordare in care se memoram totul intr-o singura tabela, dar asta ar introduce probleme de performanta precum cautarea unui fisier media in care apare o persoana, iar persoanele sunt stocate intr-o coloana ca string sub forma urmatoare "person1,person2,person3". Dupa cum se poate vedea aceasta metoda nu permite crearea de index-uri deoarece trebuie sa verificam ca "person1,person2,person3" contine persoana cautata.

2 Api-ul bazei de date

Pentru implementarea api-ului bazei de date am ales sa ma folosesc de pattern-urile Repository si Unit Of Work, iar expunerea api-ului se face prin intermediul Unit of Work-ului. Pentru implementarea unui repository am ales sa merg cu o interfata generica (IRepository) care expune metodele CRUD si o implementare generica(BaseRepository) care implementeaza acele metode CRUD.

```
public interface IRepository<TEntity> where TEntity : class
{
    TEntity GetById(Guid id);
    IList<TEntity> GetAll();
    IEnumerable<TEntity> Find(Expression<Func<TEntity, bool>> expression);
    void Insert(TEntity entity);
    void Update(TEntity entity);
    void Delete(TEntity entity);
}
```

Figura 2: Interfata repozitorului generica

Repository-ul concis pentru fiecare Entitate este derivat din BaseRepository si poate implementa metode mai specifice acelei entitati, precum media care expune o metoda de cautare dupa diversi parametri. Celelalte Repository in afara de cel de media nu au nimic in plus fata de BaseRepository. Repositoryul de media are cateva metode specifice acestuia precum:

- FindByContentType: metoda care permite cautarea unui media
- GetAllNotDeleted : metoda care returneaza fisierele media care nu au fost sterse din baza de date

GetAllNotDeleted este o metoda simpla care returneaza o lista de Media care nu au fost sterse impreuna cu toate entitatile asociate acesteia

FindByContentType este o metoda mai complicata din punct de vedere arhitectural deoarece aceasta permite adaugarea de noi optiunii de cautare fara a modifica codul de baza al api-ului, ci doar de al extinde prin intermediul dictionarelor in modul urmator Figure3

```
_searchOptions = new Dictionary<string, Func<string, Expression<Func<Media, bool>>>>();
_searchOptions.Add("any", ExpressionGenerator.SearchAny);
_searchOptions.Add("event", ExpressionGenerator.SearchByEvent);
_searchOptions.Add("people", ExpressionGenerator.SearchByPeople);
_searchOptions.Add("tag", ExpressionGenerator.SearchByTag);
_searchOptions.Add("place", ExpressionGenerator.SearchByPlace);
```

Figura 3: Salvarea expresiilor de cautare

ExpresionGenerator este o clasa care se ocupa cu generarea expresiilor. Un exemplu de metoda ar fii urmatorul Figura 4:

```
internal static Expression<Func<Media, bool>> SearchAny(string search) 1 usage
{
    return (media) => (media.Name.ToLower().Contains(search.ToLower()) ||
        media.Event.Any(e :Event => e.Name.ToLower().Contains(search.ToLower())) ||
        media.People.Any(e :People => e.Name.ToLower().Contains(search.ToLower())) ||
        media.Place.Any(e :Place => e.Name.ToLower().Contains(search.ToLower())) ||
        media.Tag.Any(e :Tag => e.Name.ToLower().Contains(search.ToLower())) ||
        media.DateOfCreation.ToString().Equals(search)) && !media.Deleted;
}
```

Figura 4: Exemplu de generare de expresie

```
public IList<Media> FindByContentType(string searchI, string type) 0+1 usages
{
    if (!_searchOptions.ContainsKey(type.ToLower()))
        throw new NotImplementedException(type);

    return _dbEntities.MediaSet.Where(_searchOptions[type.ToLower()](searchI)).ToList();
}
```

Figura 5: Apelare functiei de cautare cu expresia potrivita

Apelarea metodei FindByContentType va face urmatoarele lucruri:

- va cauta sa verifice daca are o expresie la pentru acel tip de cautare
- in cazul in care nu are va arunca o exceptie
- in cazul in care are va executa interogarea cu acea expresie

3 Declaratie

Subsemnatul Filoş Gabriel declar pe propria raspundere ca acest cod nu a fost copiat din Internet sau din alte surse. Pentru documentare am folosit urmatoarele surse:

- Carti:
- Link-uri:
 1. <https://profs.info.uaic.ro/~iasimin/>