

# Proiect Partea 1 - File Manager

**Vancsa Istvan**

## I. Descrierea Structurii Proiectului

## II. Decrierea Proiectului GUI

## III. Descrierea Proiectului Repository

## IV. Decrierea Relatiilor din Baza de Date

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

- link-uri: <https://dotnet.microsoft.com/>  
<https://docs.microsoft.com/en-us/ef/>

- **Descrierea Structurii Proiectului**

Solutia Contine doua proiecte :

- proiectul cu numele GUI de tip Windows Forms folosit pentru a crea o interfata grafica cu ajutorul caruia sa putem exemplifica functionalitatile aplicatiei
- proiectul cu numele Repository de tip Class Library(.NET Framework) care contine layerul de repository, entitatie, services si models cu ajutorul carora vom gestiona inputul oferit de catre utilizator si oferindu-i datele solicitate, astfel incat informatiile sa fie corecte, sa se mentina integritatea acestora

- **Descrierea Proiectului GUI**

Proiectul este alcatuit din 3 pagini(Form) cu ajutorul carora vom expune utilizatorului informatiile din baza de date dupa dorinta acestuia:

- **HomePage** care va pune la dispozitie utilizatorului urmatoarele view-uri:
  - 7 butoane cu ajutorul caruia utilizatorul va putea sa deschida imaginile si video-urile din 2 resurse, baza de date sau direct selectand fisierul din computer(acest fisier poate sa fie apoi adaugat in baza de date alaturi de proprietatile atribuite acestuia

- Un PictureBox folosit pentru afisarea imaginii selectate de utilizator
  - Un MediaPlayer folosit pentru afisarea video-ului selectat de utilizator
- **PhotosPage** care va pune la dispozitie o lista pentru a afisa imaginile stocate in baza de date. Fiecare element din lista va avea 3 butoane: select in cazul in care utilizatorul doreste sa vizioneze imaginea, caz in care este redirectat catre pagina Home unde o sa vada poza, modify pentru a oferi utilizatorului posibilitatea sa modifice proprietatile imaginii(sa adauge, sa stearga sau sa modifice una existenta) si un buton de delete care va permite utilizatorului sa stearga din baza de date imaginea
- **VideosPage** care va pune la dispozitie o lista pentru a afisa video-urile stocate in baza de date. Fiecare element din lista va avea 3 butoane: select in cazul in care utilizatorul doreste sa vizioneze video-ul, caz in care este redirectat catre pagina Home unde o sa vada video-ul, modify pentru a oferi utilizatorului posibilitatea sa modifice proprietatile video-ului(sa adauge, sa stearga sau sa modifice una existenta) si un buton de delete care va permite utilizatorului sa stearga din baza de date video-ul

## • Descrierea Proiectului Repository

Proiectul este alcatuit din 4 layere:

- Layerul de entities care modeleaza tabelele din baza de date respectiv contextul care reprezinta schema bazei de date
- Layerul de repositories care contine cele 4 metode C.R.U.D. care lucreaza pe baza entitatilor
- Layerul de models care este folosit la afisarea datelor din baza de date catre utilizator(nu dorim expunerea id-urilor sau a altor campuri secrete din baza de date)
- Layerul de services care va expune metode specifice fiecarui feature implementat in aplicatie, folosindu-se de layer-ul de repositories si layer-ul de models
- Layerele de repositories si de services vor avea la baza interfete pentru o abstractizare mai eficienta a acestora.

## • Descrierea Relatiilor din Baza de Date

- Baza de Date se numeste FileManager si este alcatuit din 6 tabele dupa cum urmeaza:
  - BaseEntities care este tabelul mostenit de fiecare tabela pentru mentinea unei ierarhii cat mai curate si pentru eliminarea coloanelor duplicate din baza de date
  - BaseEntities\_Movie este tabelul in care se vom stoca informatiile generale despre fiecare video

- BaseEntities\_MovieExtraProperty este tabelul in care se vor stoca informatiile particulare fiecarui video. Acest tabel este in relatie „unu la mai multi” cu tabelul BaseEntities\_Movie astfel incat fiecarei linii din BaseEntities\_Movie ii corespunde una sau mai multe linii din tabela BaseEntities\_MovieExtraProperty
- BaseEntities\_Photo este tabelul in care se vor stoca informatiile generale despre fiecare poza
- BaseEntities\_PhotoExtraProperty este tabelul in care se vor stoca informatiile particulare fiecarei imagini. Acest tabel este in relatie „unu la mai multi” cu tabelul BaseEntities\_Photo astfel incat fiecarei linii din BaseEntities\_Photo ii corespunde una sau mai multe linii din tabela BaseEntities\_PhotoExtraProperty
- BaseEntities\_Person este tabelul in care se stocheaza toate persoanele ce apar in video-uri si imagini. Acest tabel este in relatie “unu la mai multi” cu tabelele BaseEntities\_Photo respectiv BaseEntities\_Movie astfel incat: fiecarei linii din BaseEntities\_Photo ii corespunde una sau mai multe linii din tabela BaseEntities\_Person si de asemenea fiecarei linii din BaseEntities\_Movie ii corespunde una sau mai multe linii din tabela BaseEntities\_Person

Multumesc pentru timpul acordat,

Vancsa Istvan, anul III, grupa B5