MyPhotos

Subsemnata Cotet Andreea Roxana 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:

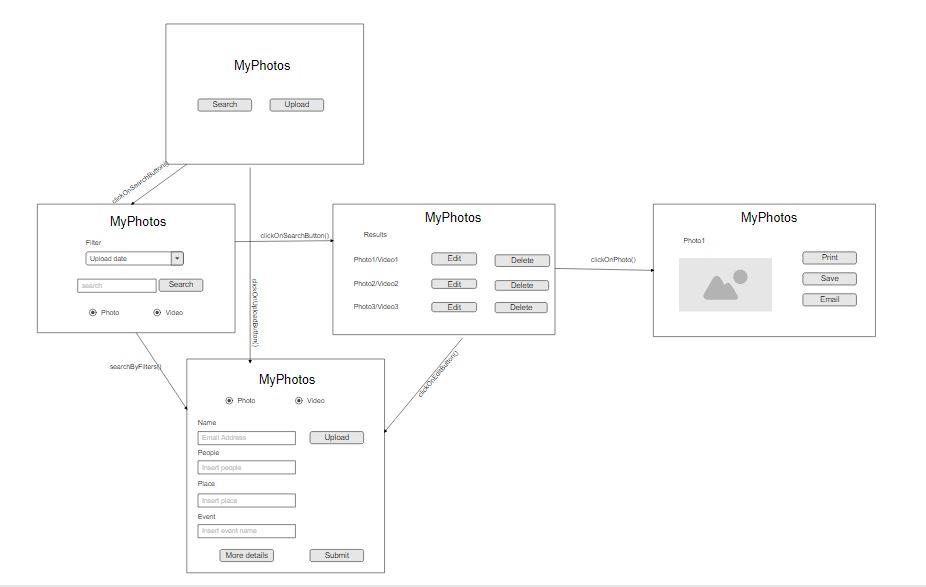
-<https://docs.microsoft.com/ro-ro/>

-<https://www.campuslife.co.in/CSharp-WindowApplication/how-to-upload-image-in-database.php>

-<https://www.dreamincode.net/forums/topic/111181-adding-video-to-an-application/>

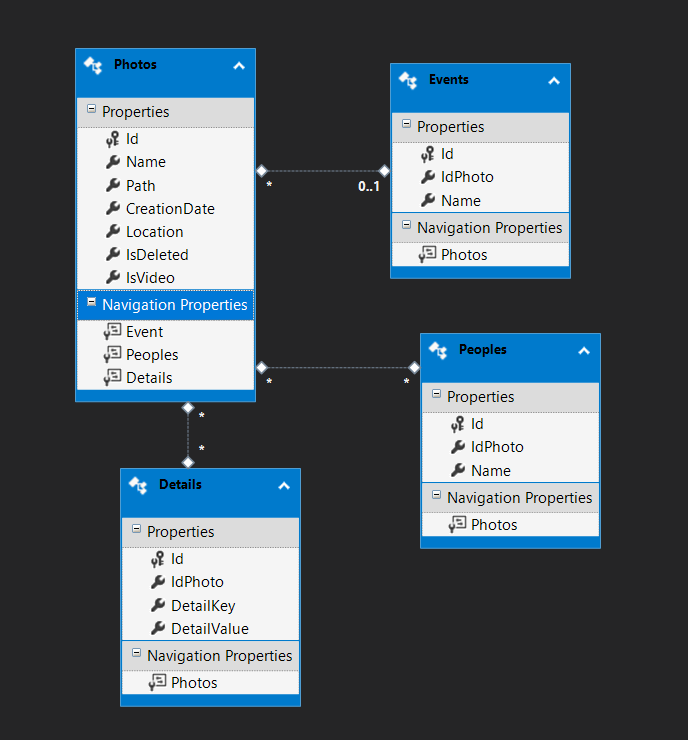
-<https://stackoverflow.com/>

In documentul prezent urmeaza sa prezint modelul pe care am decis sa-l implementez in realizarea aplictiei de gestonare a imaginilor/fimeletelor din calculatorul personal. Pentru inceput am schitat un mock-up al aplicatiei pentru o intelegere mai buna a flow-ului acesteia.

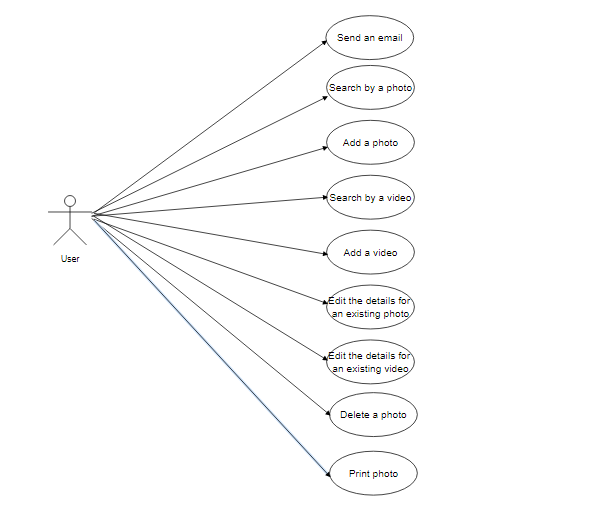


O mica descriere, utilizatorul poate alege sa caute sau sa uploadeze o imagine noua. Pentru prima optiune va trebui sa alega daca doreste sa caute o poza sau un video, apoi cautarea se poate face dupa niste filtre prestabilite cum ar fi cele mai recent adaugate, cele mai vechi, etc sau dupa cuvinte cheie cum ar fi nume de locatii, persoane care apar in poze sau tipuri de evenimente. Rezultatul cautarii vine sub forma de lista sau un mesaj de “Niciun rezultat nu a fost gasit!” in cazul in care nu exista nicio poza care sa indeplineasca conditiile cautarii. Orice poza gasit poate fi salvat extern, printat sau trimis via email. In cazul in care nu mai dorim o anumita poza stocata in baza de date, o putem sterge. Daca alegem insa sa adaugam o poza noua, ni se va deschide un formular unde vom alege poza pe care dorim sa o adaugam dar si mai multe campuri pentru diverse detalii dupa care mai tarziu putem exercita optiunea de cautare. Toate aceste lucruri sunt aplicabile si pentru video-uri.

Baza de date este una relativ simplista, o tabela pentru imagini, una pentru events, una pentru persoane si una pentru extra detalii. Exista cateva detalii standard pentru fiecare imagine sau video adaugat dar exista posibilitatea de a se adauga dinamic si altele.



Use Case:



Principalele metode folosite vor fi:

1. searchPhotoByDate(Date data, List<Photos>) - va cauta in baza de date si va returna o lista cu toate imaginile care au fost facute in data **data**
2. searchVideoByDate(Date data, List<Videos>) - va cauta in baza de date si va returna o lista cu toate video-urile care au fost facute in data **data**
3. searchPhotoByEvent(String event, List<Photos>) - va cauta in baza de date si va returna o lista cu toate imaginile care au la proprietati **event-ul** respectiv
4. searchVideoByEvent(String event, List<Videos>) - va cauta in baza de date si va returna o lista cu toate video-urile care au la proprietati **event-ul** respectiv
5. addPhoto(List<String> properties) - va adauga in baza de date o noua imagine cu **properties**
6. addVideo(List<String> properties) - va adauga in baza de date un nou video cu **properties**
7. sendEmail() - va accesa serviciu de trimitere de email-uri prin intermediul caruia se vor trimite email-uri cu o anumita poza sau un video drept content.
8. deletePhoto() - orice imagine are la proprietati un field photo\_isDeleted care by default este false atunci cand o poza este adaugata. In momentul in care se doreste stergerea unei poze, acest field este marcat cu true. Field-ul este de tip boolean.
9. deleteVideo() - orice video are la proprietati un field video\_isDeleted care by default este false atunci cand un video este adaugat. In momentul in care se doreste stergerea unui video, acest field este marcat cu true. Field-ul este de tip boolean.
10. editPhoto() - perimit eeditarea proprietilor deja existente pentru o imagine sau adaugarea de proprietati noi
11. editVideo() - perimit eeditarea proprietilor deja existente pentru un video sau adaugarea de proprietati noi