**CREAREA UNEI APLICATII CU BAZA DE DATE LOCALA IN KOTLIN**

**Elemente de M-Business**

**Ciont Cristian**

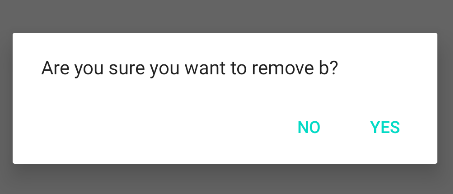
# Prezentare

Aplicatia permite utilizatorilor sa isi gestioneze jocurile video preferate care vor sa le cumpere in viitor, sub forma unui Wishlist. Acestia pot sa adauge titluri noi, sa vizualizeze lista jocurile deja existente si sa stearga orice element din lista precedenta. Toate informatiile despre entitate sunt stocate intr-o baza de date locale SQLite pentru a se pastra cand aplicatia este redeschisa.

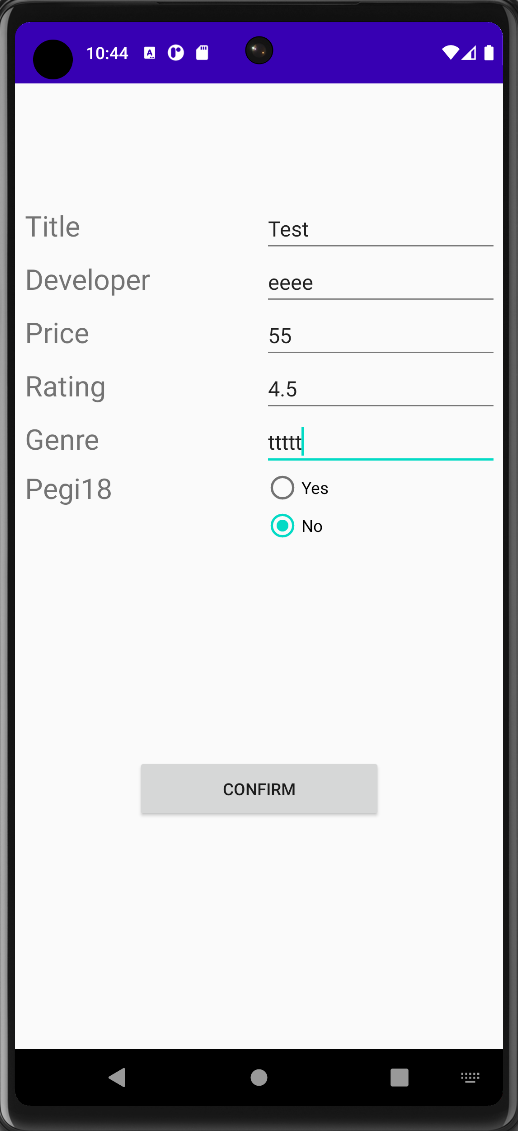
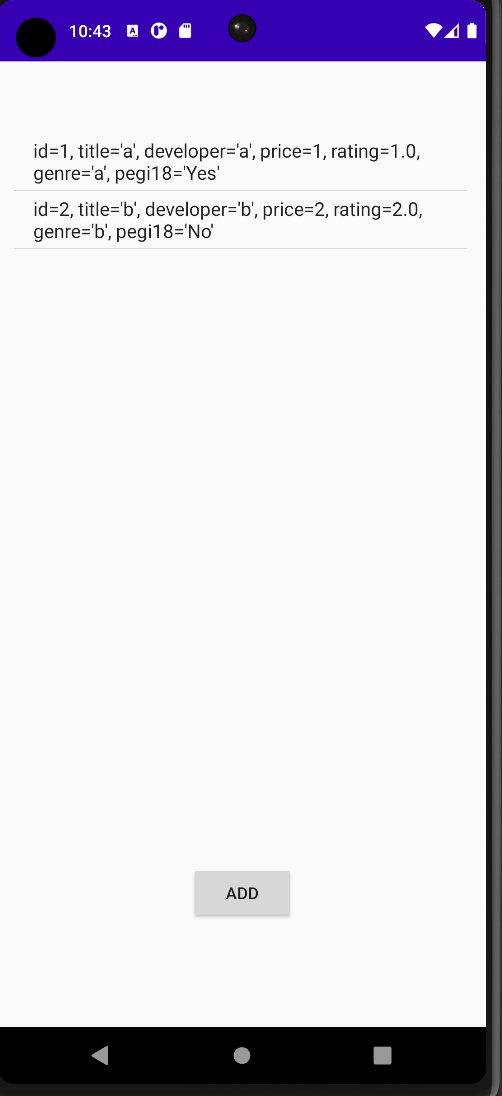
Kotlin este un limbaj de programare multiplatformă cu scop general, open-source, conceput pentru a interopera pe deplin cu Java si JVM (Java Virtual Machine). Acesta este intretinut de Google, astfel folosind Gradle ca si instrument de automatizare a construcției aplicatiilor si dependentelor.

Aplicatia este dezvoltata in Android Studio, folosind limajul Kotlin si este formata din 2 ecrane principale si un Pop-up:

1. Ecranul principal contine lista cu toate entitatile din baza de date. La un click lung pe oricare element din lista, va aparea un Pop-up de confirmare pentru stergerea elementului respectiv. De asemenea, in partea de jos a ecranului este prezent si un buton pentru adaugarea unei noi entitati;
2. Ecranul pentru adaugarea unui joc, alaturi de diferite inputs care trebuie introduse;

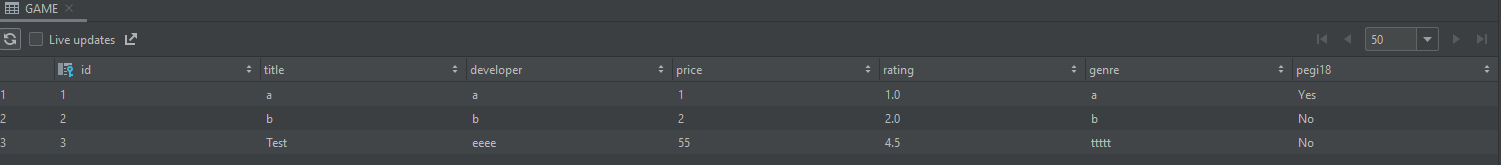


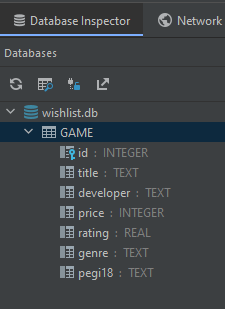
Imaginea 1: Pop-up confirmare stergere joc din lista



Imaginile 2,3 : Ecranele aplicatiei

Pentru persistarea datelor, a fost creat un tabel pentru a ingloba toate caracteristicile care trebuie salvate despre un joc. Meniul Database Inspector din cadrul Android Studio arata daca tabelele au fost create corect si ce date sunt prezente momentan.

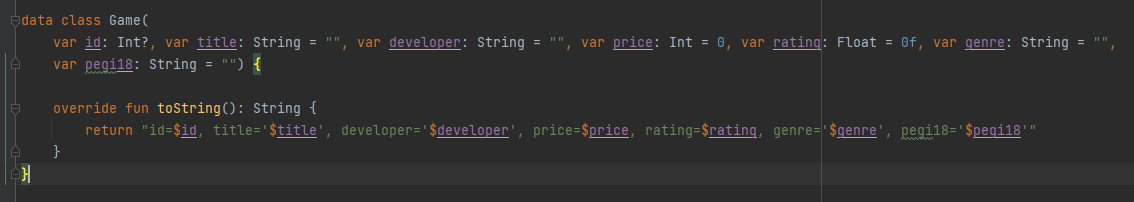
Imaginea 4: Intrarile din tabela ”Game”



Imaginea 4: Caracteristicle tabelei “Game”

# 2. Structura & Implementare

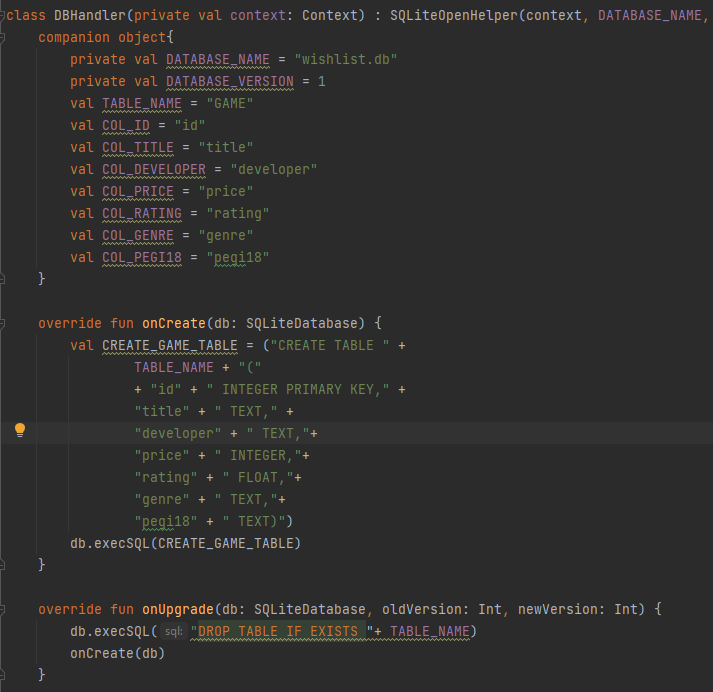
* Game.kt : Clasa model pentru entitatea Game



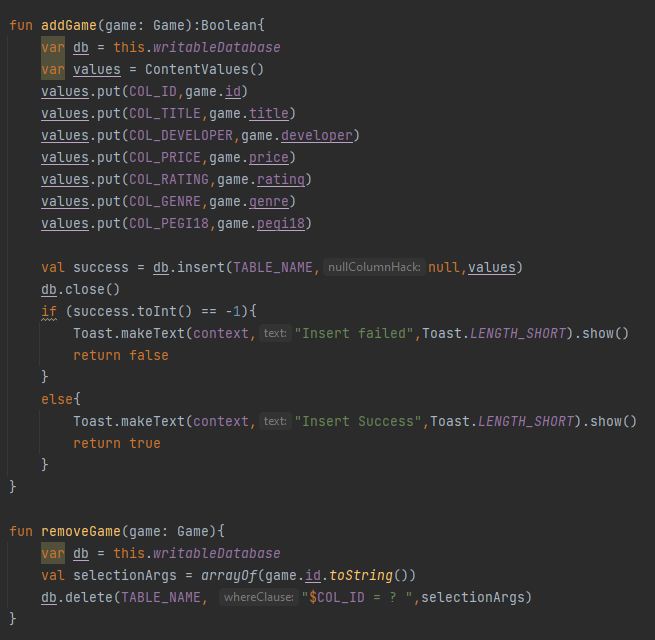
In clasa Game au fost trecute toate caracteristicile unui joc (id: Integer, title: String, developer: String, price: Double, rating: Double, genre: String, pegi18: String). Metoda toString() este folosita pentru a vizualiza obiectele in lista din ecranul principal.

* DBHandler.kt : Clasa care se ocupa cu relationarea cu baza date

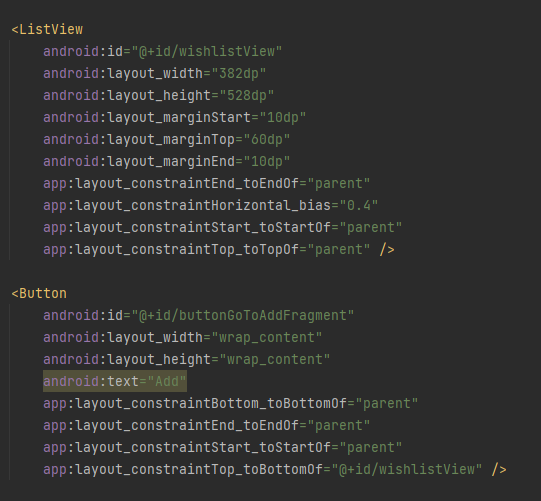
(Metode pentru crearea tabelelor, add, delete, find all)



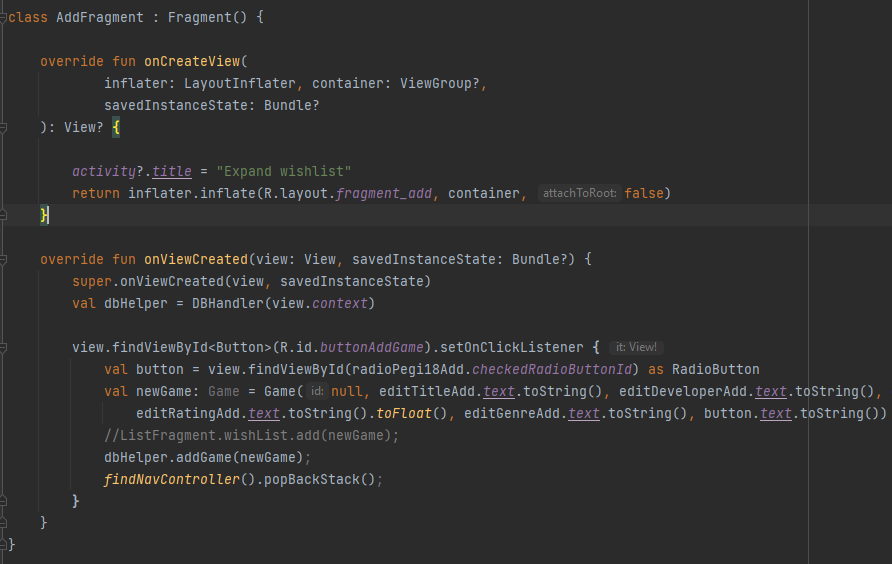
In blocul de companion object au fost declarate toate variabilele locale folosite in clasa (cum ar fi numele la tabele, denumirea pentru fiecare coloana din tabele, numele la baza de date). Metoda onCreate() executa scriptul de creare a tabelei Game, iar metoda onUpgrade() poate fi folosita cand dorim sa modificam sau sa adaugam ceva nou in structura bazei de date.



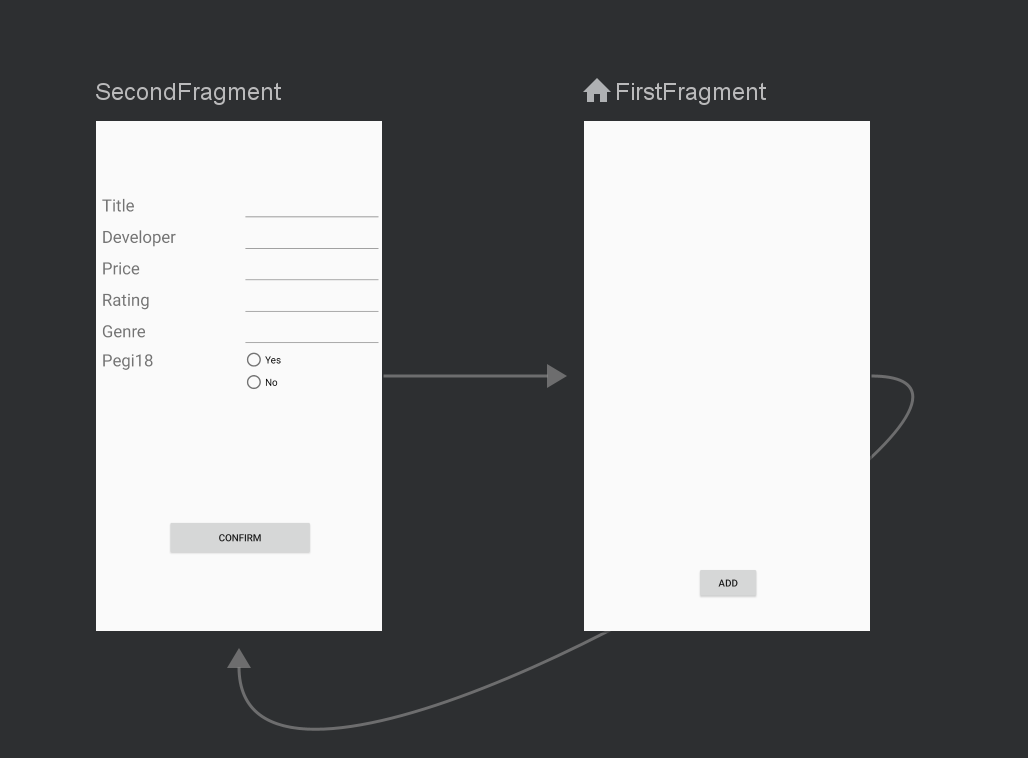
Metoda addGame(game: Game) adauga un joc nou in baza de date cu inputs specificate de utilizator si afiseaza o notificare de succes sau esec folosind libraria Toast. Metoda removeGame sterge un joc din baza de date folosind id-ul unic al acestuia.



Ecranul principal contine un Listview care este populat cu datele provenite din baza de date. In metoda OnViewCreated declaram un ArrayAdapter care are rol de intermediar intre lista de jocuri returnata de baza de date si ListView-ul care este afisat pe ecran. De asemenea, exista si un buton care in momentul apasarii ne duce la ecranul de adaugare de joc. In momentul in care un element din ListView este apasat lung, va aparea un Pop-up care cere confirmarea utilizatorului pentru stergerea elementului respectiv din baza de date.



Ecranul pentru adaugarea unui nou joc este format dintr-un tabel cu doua coloane, prima cu un label care reprezinta o caracteristica a jocului si a doua cu un EditText pentru introducerea unei valori. De asemenea, contine si un buton care in momentul apasarii trimite jocul cu toate detaliile de la utilizator catre baza de date.



Pentru tranzitia intre ecrane aplicatia foloseste un graf de navigare care este format din doua fragmente, aferente celor doua ecrane. Sagetile reprezinta metodele in care ecranele pot comunica intre ele (in cazul de fata avem o comunicare bidirectionala). O sageata de intrare inseamna ca din ecranul curent putem sa mergem catre ecranul destinatie.

# Imbunatatiri

* Adaugarea de exceptii pentru cazurile speciale ( Field-uri goale, tipuri incompatibile de variabile);
* Functii de search / filtrari pe diferite proprietati ale entitatilor
* Interfata grafica mai bogata
* Functionalitate de update a elementelor